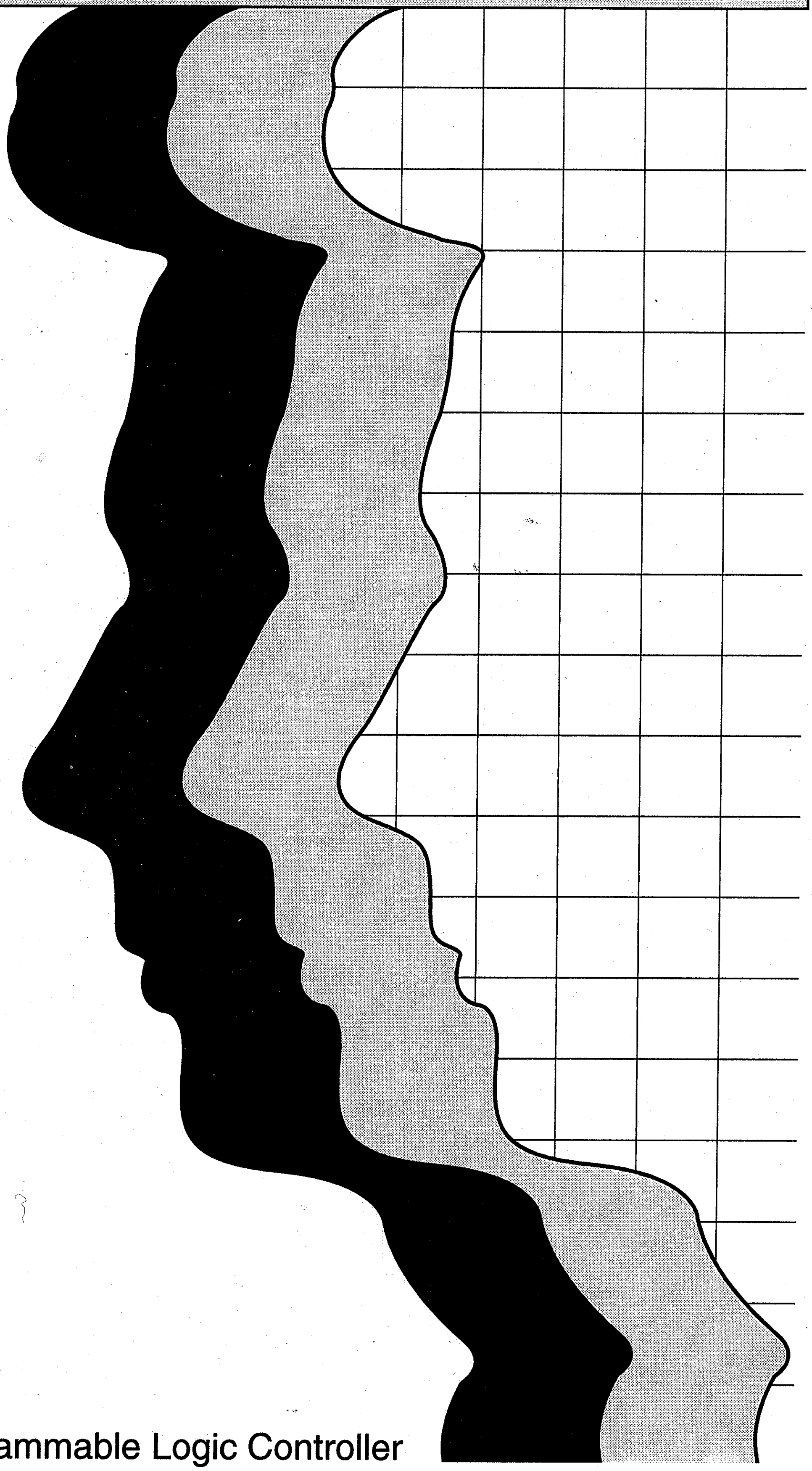
MITSUBISHI

Ethernet Interface Module
type AJ71E71-S3, A1SJ71E71-B2-S3, A1SJ71E71-B5-S3

User's Manual



A
MELSEC

Mitsubishi Programmable Logic Controller

● SAFETY PRECAUTIONS ●

(Read these precautions before using.)

When using Mitsubishi equipment, thoroughly read this manual and the associated manuals introduced in this manual. Also pay careful attention to safety and handle the module properly.

These precautions apply only to Mitsubishi equipment. Refer to the user's manual of the CPU module to use for a description of the PLC system safety precautions.

These ● SAFETY PRECAUTIONS ● classify the safety precautions into two categories:"DANGER" and "CAUTION".

DANGER	Procedures which may lead to a dangerous condition and cause death or serious injury if not carried out properly.
CAUTION	Procedures which may lead to a dangerous condition and cause superficial to medium injury, or physical damage only, if not carried out properly.

Depending on circumstances, procedures indicated by ⚠ CAUTION may also be linked to serious results.

In any case, it is important to follow the directions for usage.

Store this manual in a safe place so that you can take it out and read it whenever necessary. Always forward it to the end user.

[DESIGN PRECAUTIONS]

DANGER

- When controlling (changing data, program or operation status (remote RUN/STOP) in particular) a PLC while it is running via a device such as a personal computer connected to the special function module, configure an interlock circuit in the sequence program so that the safety of the overall system is always maintained.
  Especially, when performing the above control for a remote PLC from an external device, troubles occurring on the PLC side due to data communication error may not be handled immediately.
  Determine error handling methods between the external device and the PLC CPU for when data communication errors occur, in addition to configuring a interlock circuit in the sequence program.

⚠ CAUTION

- When installing AUI cables (transceiver cable)/coaxial cables, do not bundle them or place them close to main lines or power lines. Keep them at least 100mm (3.94 inch) away from such cables. Noise may cause erroneous operation.

[INSTALLATION PRECAUTIONS]

⚠CAUTION

- Use the PLC in the environment given in the general specifications section of this manual. Using the PLC outside the range of the general specifications may result in electric shock, fire, or erroneous operation or may damage or degrade the product.
- Make sure to switch all phases of the external power supply off when installing or placing wiring. If you do not switch off the external power supply, it will cause electric shock or damage to the product.
- Make sure to switch all phases of the external power supply off before mounting or removing the module. If you do not switch off the external power supply, it may result in electric shock, or may damage the product.
- Insert the tabs at the bottom of the module into the mounting holes in the base unit before installing the module. (Modules in AnS series, make sure screws are securely tightened to base unit with specified torques.)
  Improper installation may cause erroneous operation, failure, or the module to fall out.
- Tighten the screw within the range of specified torque.
  If the screws are loose, it may result in fallout, short circuits, or malfunction.
  Tightening the screws too far may cause damage to the screw and/or the module, resulting in fallout, short circuits, or erroneous operation.
- Do not touch the electronic parts or the module conducting area.
  It may cause erroneous operation or failure.

[WIRING PRECAUTIONS]

⚠CAUTION

- Do not connect the AUI cable when the module installation station's power is turned on.
- Be sure to fix communication cables and power cables leading from the module by placing them in the duct or clamping them. Cables not placed in the duct or without clamping may hang or shift, allowing them to be accidentally pulled, which may result in a module malfunction and cable damage.
- Perform correct pressure-displacement, crimp-contact or soldering for wire connections using the tools specified by the manufacturers. Attach connectors to the module securely.
- Tighten the terminal screws within the range of specified torque.
  If the terminal screws are loose, it may result in short circuits or malfunction.
  Tightening the screws too far may cause damage to the screw and/or the module, resulting in fallout, short circuits, or erroneous operation.
- When detaching the communication cable or power cable from the module, do not pull the cable portion. For cables with connectors, hold the connector at the junction to the module, then detach it. For connectors without connectors, first loosen the screw at the junction, then detach the cable.
  Pulling the cable portion while it is connected to the module may cause a malfunction or damage to the module and cable.

⚠ CAUTION

- Be sure that cuttings, wire chips, or other foreign matter do not enter the module.
  Foreign matter may start a fire or cause an accident or erroneous operation.

[STARTING AND MAINTENANCE PRECAUTIONS]

DANGER

- Do not touch the terminals while the electricity is on.
  Doing so could cause erroneous operation.
- Make sure to switch all phases of the external power supply off before cleaning or re-tightening screws.
  If you do not switch off the external power supply, it will cause failure or erroneous operation of the module.
  If the screws are loose, it may result in fallout, short circuits, or erroneous operation.
  Tightening the screws too far may cause damage to the screws and/or the module, resulting in fallout, short circuits, or erroneous operation.

⚠ CAUTION

- Do not disassemble or rebuild the module.
  It may cause failure, erroneous operation, injury, or fire.
- Make sure to switch all phases of the external power supply off before mounting or removing the module.
  If you do not switch off the external power supply, it will cause failure or erroneous operation of the module.

[OPERATING PRECAUTIONS]

DANGER

- Do not write data in the "system area" in the buffer memory of the special function module.
  Also, of the output signals directed to the special function module from the PLC CPU, do not output (switch on) the signals that are "use-prohibited".
  If data is written to the "system area" or output is performed with respect to a "use-prohibited" signal, it may result in the malfunction of the PLC system.

⚠CAUTION

- Before performing the control of the PLC in operation (especially changing data, program, and operation status (remote RUN/STOP)) by connecting a personal computer, etc. to the special function module, read the manual carefully and confirm if the overall safety is maintained.
  Failure to perform correct operations to change data, program, or the operation status may result in system malfunction, machine damage, or an accident.
- Remote RUN/STOP for the module installation station's PLC CPU is recommended to use the "Data Exchange during PLC CPU STOP" function after throughly reading the manual.
  If the remote RUN/STOP is executed without using the "Data Exchange during PLC CPU STOP" function, the output signal from the PLC CPU to the module goes OFF and the communication line is disconnected (close processing).
  As a result, all data transmission from other nodes, including status control of the PLC CPU, becomes impossible.

[DISPOSAL PRECAUTIONS]

⚠CAUTION

- When disposing of this product, handle it as industrial waste.

REVISIONS

* The manual number is given on the bottom left of the back cover.

Print Date	*Manual Number	Revision
Sep. 1996	SH(NA)-3598-A	First edition
Aug. 1997	SH(NA)-3598-B	$\boxed{\text{Addition}}$ SAFETY PRECAUTIONS, Item 4.7.1 CAUTION, Item 5.3.1*3, Item 5.4.5 (2), Item 10.2.8 $\boxed{1}$ Point (4), Item 10.3.6 $\boxed{1}$ Point (4), Item 13.1.1 (Error code: 7010 H), Appendix 7.2, Appendix 7.3 $\boxed{\text{Correction}}$ SAFETY PRECAUTIONS, CONTENTS, MANUALS, Chapter 1 Point, Item 1.2 $\boxed{4}$, Item 2.1, Item 2.3 $\boxed{1}$ (a), Remarks, $\boxed{2}$, Item 3.3 $\boxed{4}$, Item 3.4.1, Item 3.7.2 Point, Item 4.4, 4.5.1, 4.6.1, 4.7.2, Item 5.2 Point (1), Item 5.3.1, 5.4.2, 5.4.3, 5.4.4, Item 5.4.5 (2), Item 5.5.3, 5.6.1, 5.6.3, 7.3.1, 7.3.3, 10.1.3, 10.4.1, 10.5.1, 10.6.4, Item 13.1.1 (Error Code: A00FH), Item 13.2, Appendix 3 $\boxed{1}$ (b), Appendix 7.1, Appendix 8
Sep. 1998	SH(NA)-3598-C	$\boxed{\text{Addition}}$ SAFETY PRECAUTIONS, Chapter1 Point (2), Item 2.3 (Remark (2)), Item 3.2, Item 5.1 (*1), Item 9.2 (PLC CPU model), Item 10.4.1 (2), 10.5.3, 10.6.5 (1) (b), Appendix6, Appendix10 $\boxed{\text{Correction}}$ SAFETY PRECAUTIONS, Chapter 1 (Table), Item 1.2 (9), 1.3, 1.4, Item 2.2, 2.3 ((1)*1, (2), Point), Item 3.2, 3.3, 3.4.2, 3.5.2, 3.6.1 (Important, (3), (4), (10), (12)), 3.7.1, Item 4.2, 4.3.2, 4.4, 4.5.1, 4.7.2, 4.7.3, Item 5.1 Point, 5.3.1 (Table, (2), (11), *1, *3, point), 5.3.2, 5.3.3, 5.4.2, 5.4.3, 5.4.4, 5.5.2 (4), (7), 5.6.3, Item 6.1 (2) Point, 6.2.1 (1), 6.2.2, Item 7.1 (2) Point, 7.2.1 (1), 7.2.2 (2), 7.3.2, Item 8.2.1 (1) 8.2.2 (1) Item 10.1.1 (1) 10.1.2 (1), 10.1.3 (4), 10.6.3 (2), Item 11.1 (Point), 11.2 (1), Item 13.1.1 (7004H, A00EH),13.2, Appendix 7.2 (Program), Appendix 7.3 (Program), Appendix 8
Dec. 1998	SH(NA)-3598-D	$\boxed{\text{Correction}}$ Contents (Appendix pages), Item 1.2 (6) (9), Item 4.5.1 (2), Item 5.3.1 (page 5-10 *2 and *3), 5.4.5 (Program), Appendix 7.2 (Program), 10 (3) $\boxed{\text{Addition}}$ Contents (Appendix 11), Item 13.2 (Point), Appendix 11

Print Date	*Manual Number	Revision
June. 2000	SH(NA)-3598-E	Correction Over all (program examples), SAFETY PRECAUTIONS, Item 1.3, Item 2.3((1) Remark, (2), Point), Item 3.1, 3.2, 3.5.3, 3.6.1, 3.6.2(2)(7), 3.7.2, Item 4.4, 4.5.1, Item 5.2(Point), 5.3.1,(*1 for (7)), 5.4.1 (1)(2), 5.4.2, 5.4.3, 5.4.4, 5.5.2(5), 5.5.4, Item 6.2.3(3), Item 8.2.2, 8.2.4, Item 9.1.2(4), 9.1.3(4), 9.2*4 to *7, 9.4(7), Item 10.2.1((2), *1), 10.4.1(2), 10.6.7, Item 13.1.1, 13.2(Point) Addition Item 1.2(10), Item 3.2, Item 5.4.3(2) Remark, 5.4.4, Remark(3), Item 6.1.1 Remark, 6.1.2 Remark, 6.3.1(7), Item 7.1.1 Remark, 7.1.2 Remark, Item 8.3.1(3), Item 9.4(6), Item 13.2*1

INTRODUCTION

Thank you for purchasing the Mitsubishi programmable controller MELSEC-A Series.

Before using your MELSEC-A Series, please read this manual thoroughly to gain an understanding of the functions and performances of the A Series PLC so that the equipment is used to its optimum.

OUTLINE OF CONTENTS

CONTENTS

FIXED BUFFER EXCHANGE SECTION (CHAPTERS 6 AND 7)

RANDOM ACCESS BUFFER EXCHANGE SECTION (CHAPTER 8)

READING/WRITING DATA IN THE PLC CPU SECTION (CHAPTERS 9 AND 10)

SPECIAL FUNCTIONS SECTION (CHAPTERS 11 AND 12)

TROUBLE SHOOTINGSECTION (CHAPTER 13)

APPENDICES A- 1 to A-41

MANUALS

Following is a list of manuals related to the Ethernet Interface Module.

Related Manuals

<table>
<tr><th>Manual Name</th><th colspan="3">Manual No.
(Model Code)</th></tr>
<tr><td>Model AJ71E71-S3 Ethernet Interface Module User's Manual
(Hardware)
This manual explains the procedures for setting settings and starting up with a system configuration, unit specifications, and operation when the module is used and gives the external dimensions of the unit.
(Packaged with the AJ71E71-S3)</td><td colspan="3">IB-66687
(13J854)</td></tr>
<tr><td>Model A1SJ71E71-B2-S3/A1SJ71E71-B5-S3 Ethernet Interface Module User's Manual
(Hardware)
This manual explains the procedures for setting settings and starting up with a system configuration, unit specifications, and operation when the module is used and gives the external dimensions of the unit.
(Packaged with the A1SJ71E71-B2-S3 / A1SJ71E71-B5-S3)</td><td colspan="3">IB-66688
(13J855)</td></tr>
<tr><td>MELSECNET and MELSECNET/B Data Link System Reference Manual
This manual gives an overview and the specifications for MELSECNET (II) and MELSECNET/B and the procedures for setting the link parameters and operation and troubleshooting.
Please read this manual when accessing other stations via data link systems. (Sold separately)</td><td colspan="3">IB-66350
(13JF70)</td></tr>
<tr><td rowspan="4">MELSECNET/10 Network System Reference Manual
This system gives an overview of and the specifications for the MELSECNET/10 and the procedures for setting and operating the parameters, and explains about programming and troubleshooting.
Please read this manual when accessing remote stations via the MELSECNET/10 network system or when accessing another station using data link instructions. (Sold separately)</td><td rowspan="2">AnU</td><td>PLC to PLC</td><td>IB-66440
(13JE33)</td></tr>
<tr><td>Remote I/O</td><td>SH-3509
(13JE72)</td></tr>
<tr><td colspan="2">QnA</td><td>IB-66620
(13JF77)</td></tr>
<tr><td colspan="2">QnA/Q4AR</td><td>IB-66690
(13JF78)</td></tr>
</table>

COMMON SECTION

The common edition gives a summary of the functions and explains the features and system configuration, module specifications, and data exchange when exchanging data with the PLC CPU using a node external device via the Ethernet Interface Module.

Before using the Ethernet Interface Module, please read Chapters 1 through 5 once.

When booting up the system please follow the explanation in Chapter 4 to set the unit switch, connect with external devices, and check operations.

Abbreviated procedures for booting up the unit are given in Item 4.1.

1. GENERAL DESCRIPTION

This manual explains the Ethernet Interface Module specification, handling, and programming method for connecting the computer to the A-series PLC using Ethernet's TCP/IP or UDP/IP method.

① Model AJ71E71-S3 Ethernet Interface Module (hereafter AJ71E71-S3)

Both 10BASE5 (Ethernet) and 10BASE2 (Cheapernet) are supported and function as nodes on the Ethernet.

To switch between the 10BASE5 and 10BASE2 interfaces, use the switch on the front of the AJ71E71-S3.

② Model A1SJ71E71-B2-S3 Ethernet Interface Module (hereafter A1SJ71E71-B2-S3)

Supports the 10BASE2 (Cheapernet) and functions as a node on the Ethernet.

③ Model A1SJ71E71-B5-S3 Ethernet Interface Module (hereafter A1SJ71E71-B5-S3)

Supports the 10BASE5 (Ethernet) and functions as a node on the Ethernet.

Including these units in Ethernet makes it possible to exchange data between the A-series PLC and the computer and between QnA and A-series PLC.

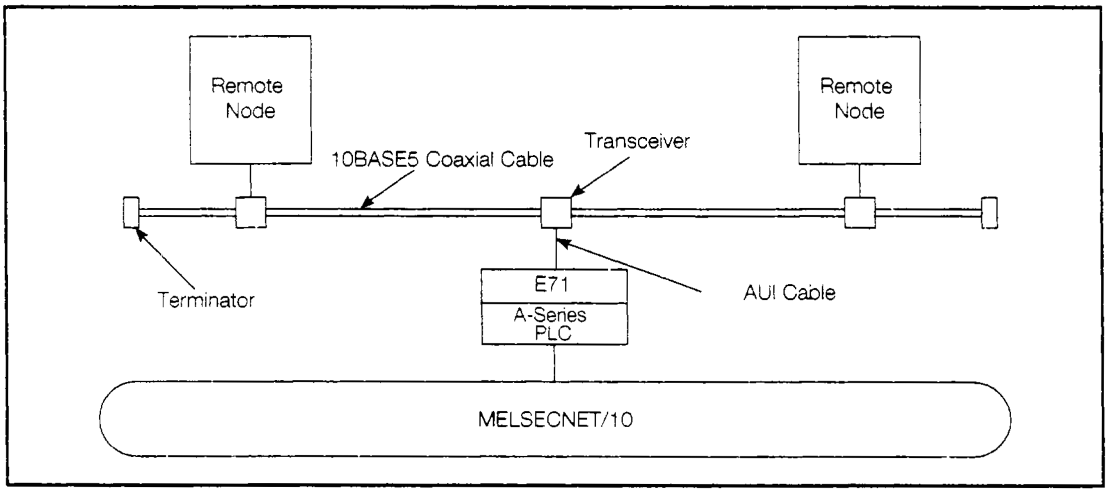


Fig 1.1 Connection Diagram Using 10BASE5 (Ethernet)

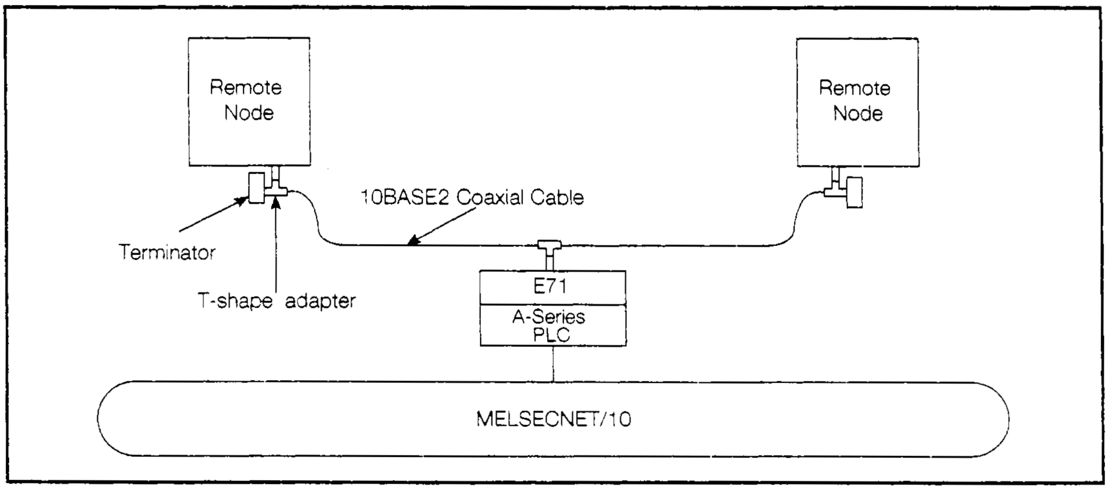


Fig 1.2 Connection Diagram Using 10BASE2 (Cheapernet)

When unpacking the Ethernet Interface Module be sure that one of either of the following products is included.

Please have the user make separate arrangements for parts and materials other than these (refer to item 2.3).

<table>
<tr><th colspan="2">Item</th><th>Product in the same package</th><th>Quantity</th></tr>
<tr><td rowspan="3">Model AJ71E71-S3 Ethernet Interface Module</td><td>Product of hardware version B or before</td><td>Main module</td><td>1</td></tr>
<tr><td rowspan="2">Product of hardware version C or later</td><td>Main module</td><td rowspan="2">1 each</td></tr>
<tr><td>Model BNC T Adapter (UG-274/U)</td></tr>
<tr><td colspan="2" rowspan="2">Model A1SJ71E71-B2-S3 Ethrnet Interface Module</td><td>Main module</td><td rowspan="2">1 each</td></tr>
<tr><td>Model BNC T Adapter (UG-274/U)</td></tr>
<tr><td colspan="2">Model A1SJ71E71-B5-S3 Ethernet Interface Module</td><td>Main module</td><td>1</td></tr>
</table>

Point

(1) This manual explains the functions and methods of use for the Ethernet Interface Modules AJ71E71-S3, A1SJ71E71-B2-S3, and A1SJ71E71-B5-S3. In regards to the method for connecting to Ethernet, please read the explanations corresponding 10BASE2 or 10BASE5.

(2) Even though hardware specifications for the switches, 10BASE2 connector and other areas of AJ71E71-S3 have been changed in hardware version C or later, the functional and performance specifications in those areas are the same as the conventional products. The user can use the product just in the same manner as the conventional one.

(3) In this manual the general terms for the Ethernet interface module will be E71, 10BASE5, and 10BASE2; and the general term for network will be Ethernet.

1.1 Software Configuration

E71 supports the TCP/IP and the UDP/IP protocols.

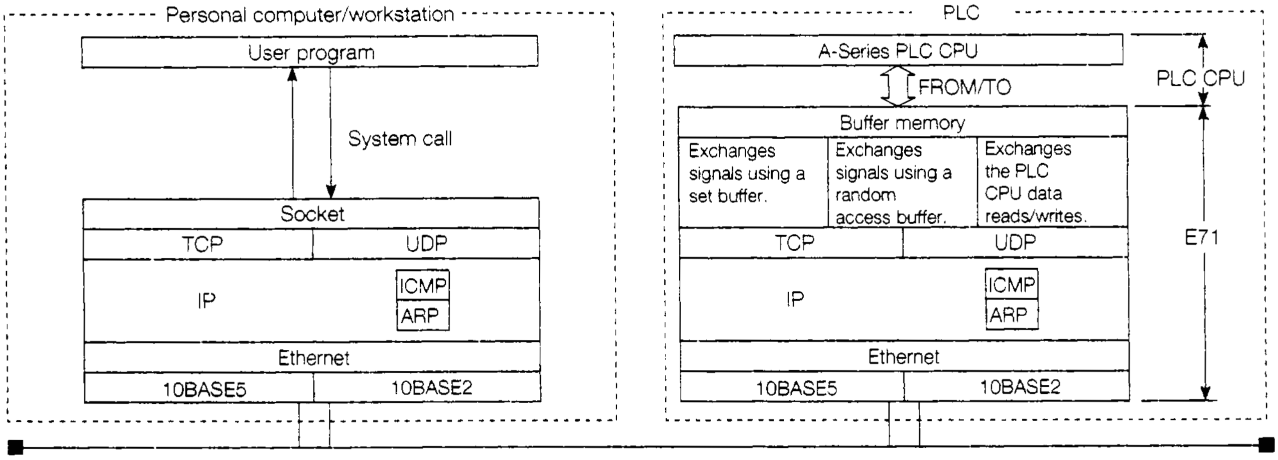


Fig 1.3 Software Configuration Diagram

1 TCP (Transmission control protocol)

This protocol retains the data reliability and correctness for the TCP protocol level.

- Establishing a connection creates a theoretical connection as if a special line were created between the nodes.
- A maximum of 8 connections can be established at the same time and communication to multiple buffers can be done at the same time.
- Data reliability is maintained by using a check sum for PLC control and data read transmit functions using the PLC No.
- The communication data flow can be controlled using window operations.
- Supports a MAX SEGMENT option.

2 UDP (User datagram protocol)

This protocol retains the data reliability and correctness on the UDP protocol level.

However if the data does not reach the target node it will not be retransmitted.

- Because it is connectionless, high speed transmission is possible.
- A check sum is used to increase the reliability of the transmitted data. However when greater reliability must be maintained, use a user application or TCP.

3 IP (Internet protocol)

- Data transmissions can be sent and received using the datagram format.
- The transmitted data can be divided and reassembled.
- Routing options are not supported.

4 ARP (Address resolution protocol)

- A broadcast is used to find the Ethernet physical address from the IP address.

5 ICMP (Internet control message protocol)

- Has a function to transmit IP error messages.
- Please refer to the attachment for information regarding the ICMP option support type (ICMP protocol).

Ethernet is the registered trademark of XEROX CO., LTD.
10BASE2 is the formal way to say Cheapernet.
There is no registered trademark for Cheapernet.

1.2 Features

E71 is a unit used to connect the A-series PLC to the Ethernet.

By combining a A-series PLC in the Ethernet it is possible to construct a network system that combines data link system/network with the Ethernet.

It is possible to conduct fixed buffer communication with a remote node and to read and write data from the random access buffer exchange area from the PLC CPU.

Fixed buffer exchange using TCP/IP or UDP/IP, random access buffer exchange, and reading and writing data inside the PLC CPU (general data exchange) from a remote node is possible.

The main features of the E71 are explained below.

1 **Selecting the exchange format and exchange node units is possible (see Chapter 5 for a detailed explanation)**

① Whether to use the TCP/IP or the UDP/IP communication protocols can be selected for each remote node that exchanges data, and the communication line for the target remote node can be set to open (communication line connect).

② Eight communication lines can be open at the same time and data can be exchanged with multiple remote nodes.

③ The relationship between the E71 data exchange and the selectable exchange formats is shown below.

Exchange Function		TCP/IP	UDP/IP
Exchanging using a fixed buffer	With procedures	○	○
	Without procedures	○	○
Exchange using a random access buffer		○	○
Reading/writing data inside the PLC CPU using requests from a remote node (General data exchange)		○	○

2 Data exchange while the PLC CPU is stopped is possible (Detailed explanation in Item 5.6).

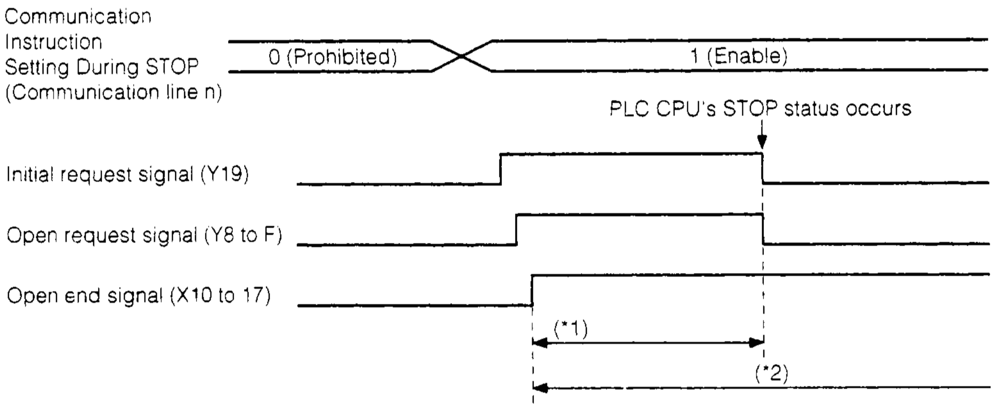
① When conducting the following data exchange, the data exchange with remote nodes can be continued even if the PLC CPU installed in the E71 is stopped after the communication line is opened by the PLC program.

(Function that makes it possible to continue exchange when in STOP status)

- Exchange using random access buffer
- Writing/reading data inside the PLC CPU with a request from a remote node (general data exchange)

  * In either case exchange can be continued using the communication protocol at the time the communication line is opened.

② Data exchange while the PLC CPU is stopped is conducted by setting the buffer memory's Exchange Instruction Area During STOP (address: 496) to enable. (Set for each communication line.)



*1 Exchange possible range when the exchange instruction during STOP is set to prohibit.
*2 Exchange possible range when the exchange instruction during STOP is set to enable.

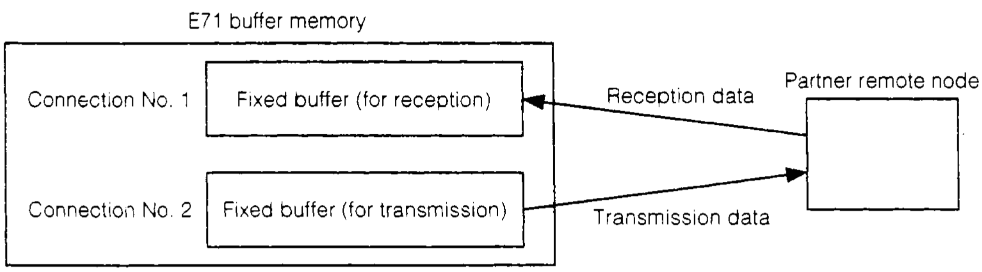
3 Data exchange on a 1:1 or 1:n with each remote node (exchange using the fixed buffer)

(a) This conducts data exchange between the remote node and the PLC CPU on a 1:1 (TCP/UDP) or a 1:n (UDP only) basis using the E71's fixed buffer.

(b) The E71 has 8 fixed buffers with a memory capacity of 1k words, and the partner remote node with which exchange will be done, application (transmission/reception), and the protocol to be used (TCP/UDP) can be set for each fixed buffer. (Exchange between an E71 and another E71 is possible.) When conducting transmission and reception with the same remote node, 2 fixed buffers are required.

(c) Setting pairing open using communication line open processing creates a pair containing a reception fixed buffer and a transmission fixed buffer and connects the partner remote node with 1 port through which data can be exchange. (Detailed explanation in Item 5.4.4)

(Example)



(d) When exchanging with fixed buffers, exchange can be done using either the E71 procedure (with procedure) or without procedures.

(Please refer to Item 3.3 for details regarding the amount of data that can be transmitted at one time.)

① Exchanging when there is with procedure (Refer to Chapter 6 for details.)

The E71 protocol transmits and receives data on a 1:1 basis using a handshake between the specified node and the PLC CPU.

Use when transmitting or receiving simple data from the PLC program.

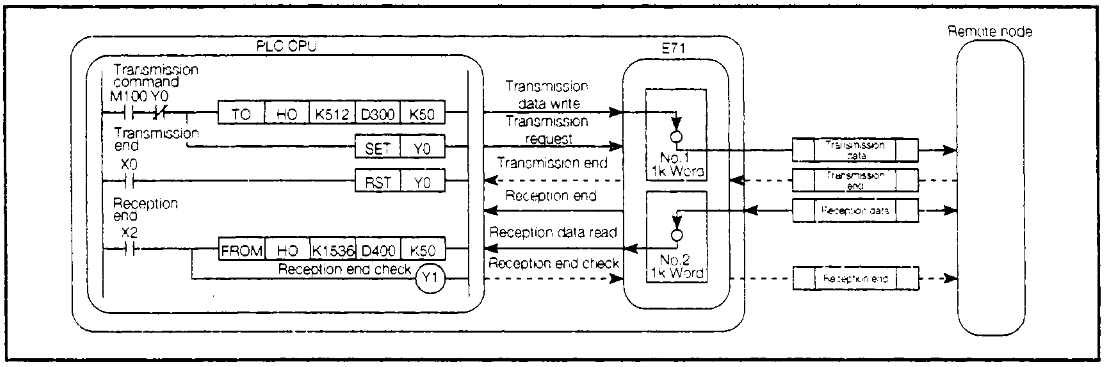


Fig 1.4 Fixed Buffer Exchange (With Procedure)

- When transmitting data

  After writing the transmission data in the fixed buffer, the transmission request signal will turn on and data of the specified number of words will be transmitted.

  When a transmission end response is received from the remote node the transmission end signal will turn on.

- When receiving data

  When the data of the specified number of words is stored in the reception fixed buffer, the reception end signal will turn on.

  When the reception data is read from the fixed buffer and the reception end check signal is turned on, the reception end check response will be sent and the reception end signal will be turned off.

② When exchanging without procedure (Detailed explanation in Chapter 7)

Data transmission is conducted when the specific node and the PLC CPU are 1:1 or 1:n by simultaneous broadcast communication (Simultaneous broadcast communication function, refer to [4]).

Used to transmit fixed buffer data or to put reception data in the fixed buffer the way it is received.

Because it is without procedure the handshake with the remote node must be done by the sequence program.

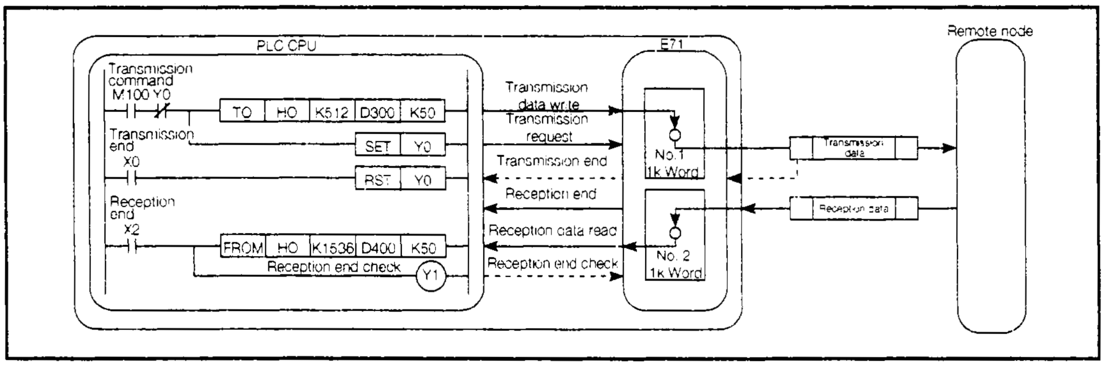


Fig 1.5 Fixed Buffer Exchange (Without Procedure)

- When transmitting data

  After the transmitted data is written in the fixed buffer and the transmission request signal is turned on, the data of the specified number of words is transmitted and the transmission end signal is turned on.

  The transmission end response does not wait for a reception signal.

- When receiving data

  When the data of the specified number of words is received and stored in the fixed buffer, the reception end signal is turned on.

  When the reception data is read from the fixed buffer and the reception end check signal is turned on, the reception end signal turns off.

  The reception end response does not wait for a transmission signal.

[4] Simultaneous broadcast communication (Details explained in Item 7.3)

User for simultaneous broadcast to all remote nodes on the same Ethernet that is connected to the E71 using the UDP/IP functions, refer to fixed buffer exchange without procedure.This makes it possible to write the same data. However, the remote node must be performed read and delete processing when received message is not required by this simultaneous broadcast communication.

5 **Data exchange from a read/write request from a remote node (Random access buffer exchange)**

Use when processing the maximum 6k word data quantity with the sequence program and when processing transmission and reception data when the sequence program and the remote node are non synchronous (Detailed explanation in Chapter 8).

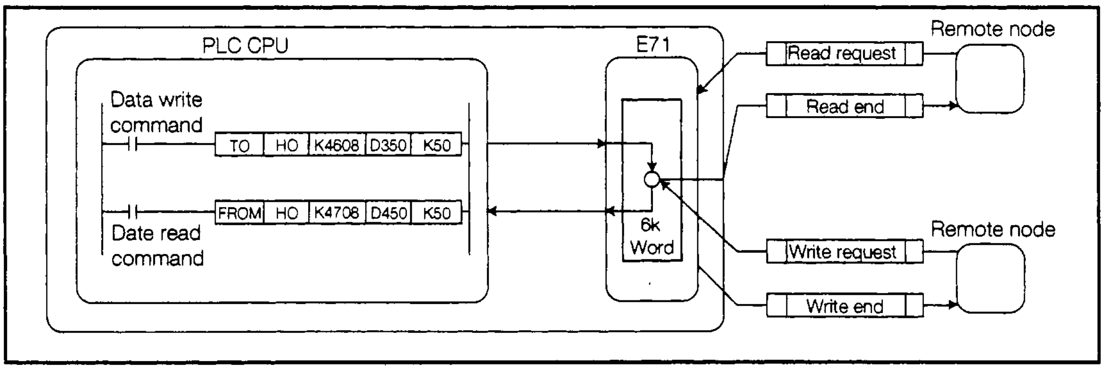


Fig 1.6 Random Access Buffer Exchange

(a) The random access buffer can read and write the same address to multiple nodes. However, exchange between the PLC CPU and the remote node is non synchronous so the user must conduct interlock processing.

(b) The random access buffer holds 6k words (3k for channel 0 and 3k for channel 1).

The area is not set for each connection as for fixed buffer exchange.

(c) The PLC CPU reads and writes data to and from the random access buffer by switching channels in the 3k word unit.

However, communicating nodes use this buffer area as one continuous area of 6k words.

(For details refer to Item 3.3)

(d) The PLC CPU processing when data is transmitted and received is as follows.

- When transmitting data

  The transmitted data can be written into any area of the random access buffer.

  When a read request is received from a remote node the data written in the specified area of the random access buffer and an end response is transmitted.

- When receiving data

  When a write request is received from a remote node the received data is stored in the specified area of the random access buffer and a write end response is transmitted.

  The received data is read from the random access buffer.

(e) Writing to and reading from the random access buffer from a remote node can be freely done between nodes set in the E71 parameter.

For this reason, the random access buffer can be used to store common data and to receive and relay data between remote nodes (There is no need to use the PLC CPU memory).

6 **Reading and writing inside the PLC CPU via a request from a remote node (general data exchange : detailed explanation in Chapters 9 and 10)**

Use to read and write remote station PLC data via local station PLC installed in the E71 and data link systems/network systems from the remote node with MITSUBISHI MELSEC communication support software tool (*1) inserted. In addition, the PLC CPU state can be controlled from a remote node using remote RUN/STOP, etc.

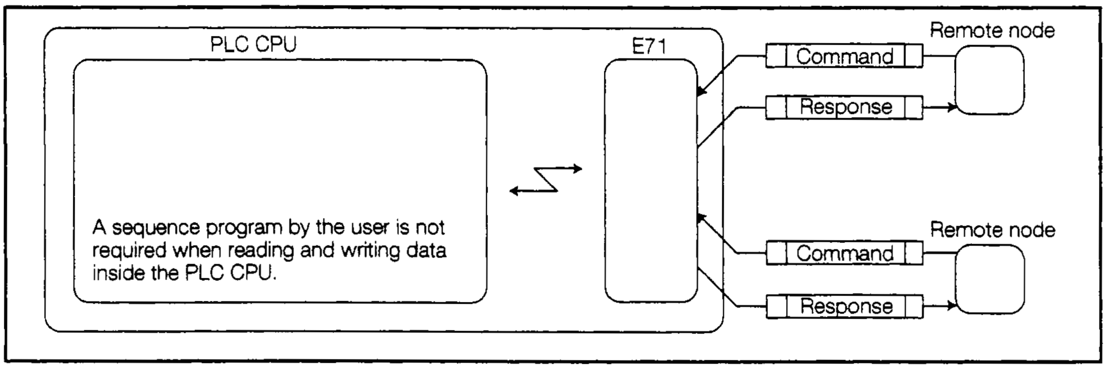


Fig 1.7 Reading and Writing the PLC CPU

(a) When a remote node transmits a request for a read/write of data inside the PLC CPU to the E71, the data in all devices, programs, commands, and parameters is transmitted to or received from the E71.

(b) When the PLC CPU installed in the E71 is connected to a data link system or a network system, data can be exchanged to and from the remote PLC CPU. (For details refer to Item 9.1.)

(c) Because all data exchange is conducted between the E71 and the remote node, data exchange can be conducted by having the sequence program only conduct initial processing and communication line open processing.

It is not necessary to create a special sequence program to exchange data.

*1 Communication programs in the personal computers to be connected to Ethernet or computer link can be simplified by using the following communication support software tools manufactured by Mitsubishi Electric, which support communication between MELSEC-A or QnA series PLC and personal computers.

- SWnD5F-CSKP-E type basic communication support tool

The overview and application examples of basic communication support tools are shown in Section 11 in the Appendix. Refer to it as needed.

7 **Selecting the exchanged data's data code (Details explained in Item 3.3)**

Use the following functions to select an exchange data code (ASCII code/binary code) that matches that of the remote node when exchanging data between the E71 and the remote node.

- Fixed buffer exchange buffer exchange with procedure
- Random access buffer exchange
- Reading/writing data from the PLC CPU using a request from a remote node

Code conversion during exchange is conducted by the E71 and all data received between the E71 and PLC CPU is in binary code.

For this reason, a sequence program for code conversion is not required. However, the selection of exchange data codes is done on a unit basis so selection for each port cannot be done.

8 **Router relay function (Details explained in Chapter 12)**

Used when exchanging by relaying the router.

This function does not operate as a router but is a function to make exchange via routers and gateways.

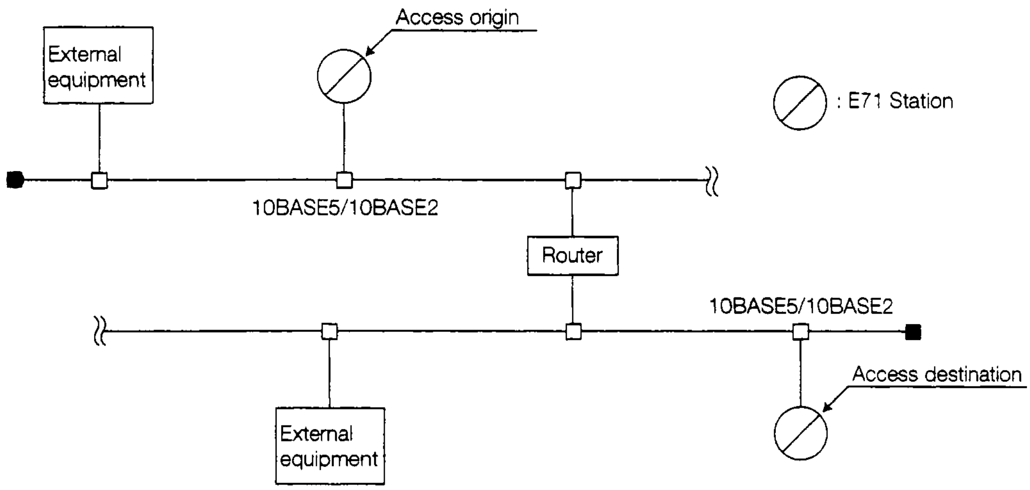


Fig 1.8 Router Relay Function

9 **Other remote node existence check function (Details explained in Item 5.3.1)**

Used to have the E71 regularly check if the other remote node for which the connection was made exists.

When exchange has not been conducted with the other node for a specified period of time the E71 checks whether the node is operating properly.

When the other node is not operating properly the E71 closes the line (connection forced disconnect).

10 **Accessing a PLC CPU from a peripheral device for the GPP function through Ethernet connection (Details explained in the GPP Manual)**

Access can be made to other station's PLC CPU on the MELSECNET/10 or MELSECNET (II) /B via the ACPU of a station equipped with an E71 or via a station equipped with an E71 through Ethernet connection from a peripheral device (*1) for the GPP function.

*1 Indicates a peripheral device for GPP in which the following GPP function software is installed:

GPPW: GPP Function Software (Product after SW2D5C/F-GPPW-E)

1.3 Comparison with AJ71E71

The differences in the specifications between the E71 and the previous Ethernet interface modules (AJ71E71, A1SJ71E71-B2 (B5)) are shown below.

Functions for which a circle is placed in both the AJ71E71 column and the E71 column shows compatibility. (However, a some of the communication timing varies, so the response timeout time must be adjusted.)

	Date Exchange Functions/Specifications		AJ71E71 (Previous products)	E71 (This product)	Remarks
1	Communication protocol selection function for the partner remote node unit		○	○	————
2	Fixed buffer exchange	With procedure	○	○	————
		Without procedure	×	○	
3	Exchange using pairing open		×	○	For fixed buffer exchange
4	Simultaneous broadcast exchange		×	○	Exchange is possible using a fixed buffer without procedures (UDP/IP open is possible)
5	Random access buffer memory exchange		○	○	————
6	Data read/write in the PLC CPU		○	○	General data exchange function
7	Exchange while the PLC CPU is stopped		×	○	Exchange is possible after the port is opened regardless of the PLC CPU's RUN/STOP status.
8	Exchange data code (ASCII code/binary code) selection		○	○	————
9	Router relay exchange		×	○	Static router relay
10	Partner remote node existence check		×	○	————
11	Timer setting value units for data exchange	500ms	×(*1)	○	Timer value units to be set during initial processing
		2s	○ (Fixed)	○	
12	COM.ERR LED turned on/off notification		×	○	I/O signal with the PLC CPU (X1B)
13	Connection of the peripheral device for GPP function (Products after SW2D5C/F-GPPW-E)		×	○	Connection via Ethernet

*1 When the module software version is before the Q version.

Remarks

When utilizing the remote node program for data communication with previous Ethernet interface modules for data communication with this E71, refer to the appendices.

1.4 Terms, Abbreviations, and Terminology Used in This Manual

1 Module terms and abbreviations

This manual uses the following terms and abbreviations for the Ethernet interface module and the PLC CPU unit. When display of the model name is required, the module name will be returned.

Abbreviations/Terms	Descriptions/Pertinent Unit
ACPU PLC CPU	The appropriate CPU module shown in Item 2.2. Sometimes shown as CPU in diagrams. (Including PLC CPUs with MELSECNET data link functions)
AnACPU	A2ACPU, A2ACPU-S1, A3ACPU, A2ACPUP21/R21, A2ACPU21/R21-S1, and A3ACPUP21/R21 in ACPU
AnUCPU	A2UCPU, A2UCPU-S1, A3UCPU, A4UCPU, A2ASCPU, and A2ASCPU-S1 in ACPU
QnACPU	Q2ACPU, Q2ACPU-S1, Q3ACPU, Q4ACPU, Q4ARCPU, Q2ASCPU, Q2ASCPU-S1, Q2ASHCPU and Q2ASHCPU-S1 in ACPU
AnA/AnU/QnACPU	AnACPU, AnUCPU, QnACPU
AnU/QnACPU	AnUCPU, QnACPU
E71	AJ71E71-S3, A1SJ71E71-B2-S3, A1SJ71E71-B5-S3
A1SJ71E71-B2/B5-S3	A1SJ71E71-B2-S3, A1SJ71E71-B5-S3
AJ71E71	AJ71E71, A1SJ71E71-B2, A1SJ71E71-B5 (Previous products)
QE71	AJ71QE71, AJ71QE71-B5, A1SJ71QE71-B2, A1SJ71QE71-B5
QLP21/QBR11	AJ71QLP21 (S/G), AJ71QBR11, A1SJ71QLP21, A1SJ71QBR11
LP21/BR11	AJ71LP21 (G), AJ71BR11, A1SJ71LP21, A1SJ71BR11
QLP25/QBR15	AJ72QLP25 (G), AJ72QBR15, A1SJ72QLP25, A1SJ72QBR15
LP25/BR15	AJ72LP25 (G), AJ72BR15

2 Other terms and abbreviations

This manual uses the following terms and abbreviations to explain the E71 data exchange functions. When it is necessary to clearly show what is being explained the name or model name will be written.

Abbreviation/Terms	Description
External equipment	Remote node personal computers, computers, workstations (WS), and other E71/AJ71E71/QE71s, etc. connected to the Ethernet to exchange data.
Data link system	MELSECNET (II), MELSECNET/B data link systems
Data link module	MELSECNET (II), MELSECNET/B modules
Network	10BASE5, 10BASE2, network system, data link system
Network system	MELSECNET/10 network system
Network module	MELSECNET/10 module
Ethernet	10BASE5, 10BASE2
I/F	Interface
MELSECNET	Network system, data link system
Local station	E71 installed station's PLC

3 Terminology

For information on terminology, please use the index provided at the end of the appendixes in this manual.

2. SYSTEM CONFIGURATION

This section explains the system configurations that are possible in combination with the E71.

2.1 Overall Configuration

Following shows a system configuration with an E71 PLC installed in the Ethernet.

Please refer to Item 2.3 for information regarding other arrangements that must be made by the user.

1 When connecting the PLC CPU with the Ethernet

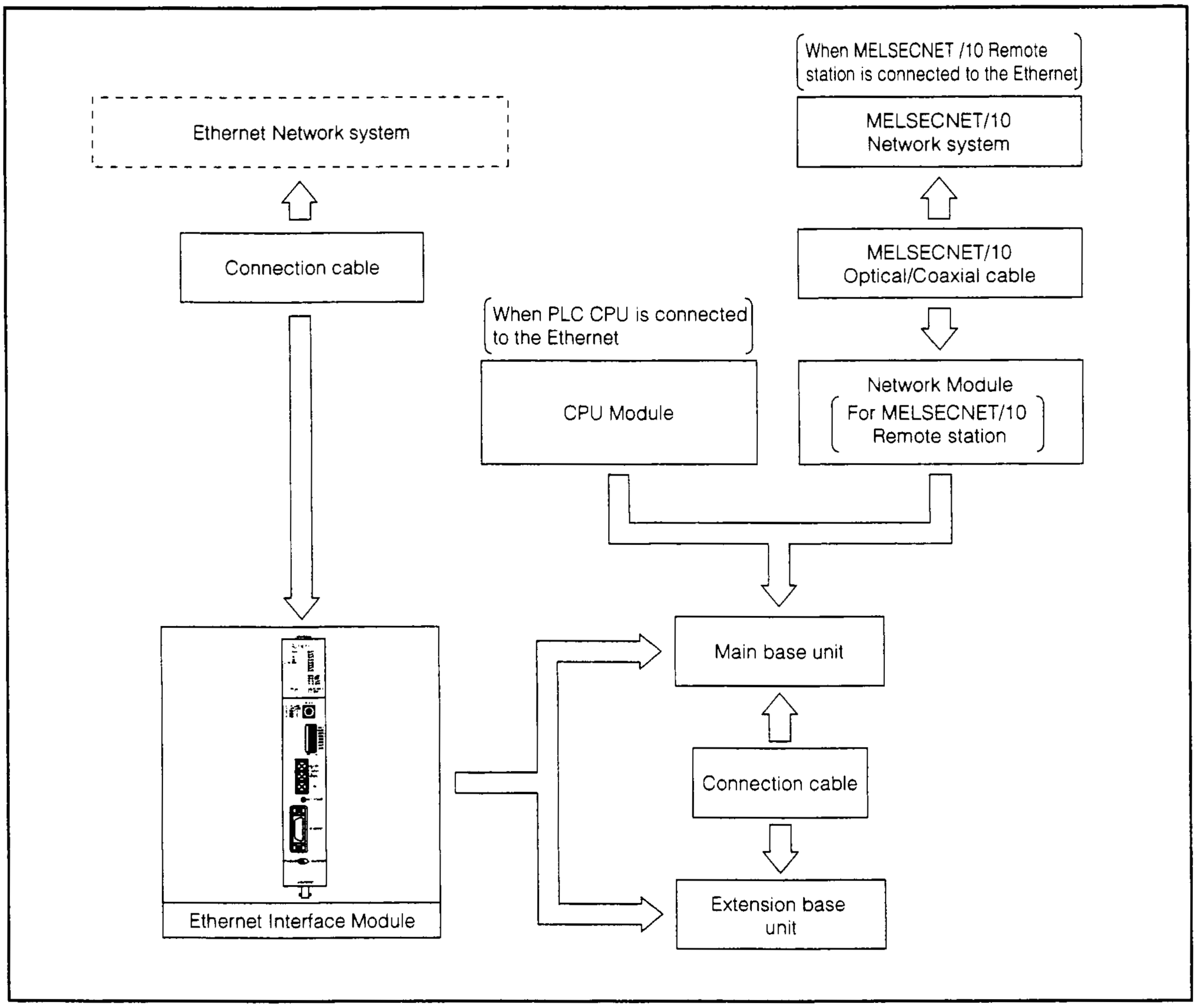


Fig 2.1 System Configuration Diagram

2.2 Supported Systems

The E71 can be used for the system described below.

1 Supported units and installable number of units

The following table shows the PLC modules that can be installed in the E71 and the number that can be installed.

<table>
<tr><th>E71 installation station</th><th>Application module</th><th>Number of modules that can be installed</th><th>Remarks</th></tr>
<tr><td rowspan="3">CPU module</td><td>A0J2H
A1, A1N
A1S(-S1), A1SJ
A1SH, A1SJH
A2(-S1)
A2N(-S1)
A2S(-S1)
A2SH(-S1)
A3, A3N</td><td>2</td><td rowspan="4">When using both the AnS series and A series special function modules GOT series shown below, the total number of modules that can be installed includes the number of these modules used and connected.

· A1SJ71UC24-R2 · A1SJ71C24-R2
· A1SJ71UC24-R4 · A1SJ71C24-R4
· A1SJ71UC24-PRF · A1SJ71C24-PRF
· A1SD51S
· A1SJ71E71-B2/B5-S3 · A1SJ71E71-B2/B5
· AD51(-S3) · AD51H(-S3)
· AD51FD(-S3)
· AD57G(-S3)
· AJ71C21(-S1) For only the BASIC program
· AJ71C23(-S3)
· AJ71UC24 · AJ71C24(-S3/-S6/S8)
· AJ71P41
· AJ71E71-S3 · AJ71E71
· A0J2C214-S1
· GOT series (Only when bus connection)

* When using a computer link module (AJ71UC24, etc.) as a multiple drop link module, it is not included in the above restrictions on the number of modules that can be installed. Multiple modules can be installed within the PLC CPU's I/O number of points</td></tr>
<tr><td>A1SCPU-C24</td><td>1</td></tr>
<tr><td>A2A(-S1)
A3A
A2U(-S1)
A3U
A4U
A2AS(-S1)
Q2AS(-S1)
Q2ASH(-S1)
Q2A(-S1)
Q3A
Q4A, Q4AR</td><td>6</td></tr>
<tr><td>MELSECNET/10 (Remote station)</td><td>AJ72LP25
AJ72BR15
AJ72LR25
AJ72QLP25
AJ72QBR15
A1SJ72QLP25
A1SJ72QBR15</td><td>2</td></tr>
</table>

2 Installable base units

Except where noted below the basic base unit and the extension base unit can be freely used in installation slots of the E71.

(a) Installing the extension base unit (models A52B, and A55B) without a power unit could make the amount of power supply insufficient, so doing so should be avoided as far as possible. If this unit is installed be sure to give sufficient consideration to the current capacity of the main base unit's power unit, and the extension cable voltage drop when selecting the extension cable. (For details to the usable CPU unit's users manual) (Refer to (1).)

(b) The E71 can be installed in the PLC CPU based unit and the MELSECNET/10 remote station. It cannot be installed in MELSECNET (II) and MELSECNET/B remote stations.

3 **Accessible PLC**

This shows the remote station PLC that can be accessed via an E71 installable station from a remote node. Each accessible CPU unit has a unit that includes MELSECNET link functions.

(Example) In the case of the A3ACPU, the A3ACPUP21 and A3ACPUR21 can be accessed.

① PLC CPU

PLC CPUs that can be accessed from remote nodes can also be accessed via data link systems and network systems. For access refer to Item 9 and Item10.

PLC CPU	A0J2H	A1	A1N	A1S(S1)	A1SJ	A1SH	A1SJH
	A2(S1)	A2N(S1)	A2A(S1)	A2U(S1)	A2S(S1)	A2SH(S1)	A2AS(S1)
	A2C	A2CJ	A3	A3N	A3A	A3U	A4U
	Q2A(S1)	Q2AS(S1)	Q2ASH(S1)	Q3A	Q4A	Q4AR	

② Remote station

Shows the remote stations that can be accessed from a remote node via the data link system and network system. The buffer memories of the special functions units of the remote stations that are connected by the link units that are named below can be accessed.

MELSECNET/10	AJ72QLP25	AJ72QBR15	A1SJ72QLP25	A1SJ72QBR15	AJ72LP25(G)
	AJ72BR15	AJ72LR25			
MELSECNET (II)	AJ72P25	AJ72R25			
MELSECNET/B	AJ72T25B	A1SJ72T25B			

2.3 Devices Required for Network Configuration

1 **The equipment shown in Figure 2.2 are required when connecting to 10BASE5. The user will please make the arrangements.**

(a) Only use 10BASE5 coaxial cable, N-type connectors, N-type terminators, transceivers, AUI cable (transceiver cable) that meet Ethernet standards. Please use transceivers that have signals that are generally called SQETEST or Heartbeat (transceiver function that uses a signal to check if the transceiver is operating correctly after transmission).

(SQETEST: Signal Quality Error TEST)

<table>
<tr><th></th><th>10BASE5</th></tr>
<tr><td>Transmission medium</td><td>Coaxial cable (Ethernet standard cable)
50 Ω</td></tr>
<tr><td>AUI cable
(Transceiver cable)</td><td>Twisted pair cable with 15 pin D connector
*E71s 10BASE5 connection connector layout
<table>
<tr><th>Pin No</th><th>Signal Name</th><th>Pin No</th><th>Signal Name</th></tr>
<tr><td>1</td><td>FG</td><td>9</td><td>Collision detection (−)</td></tr>
<tr><td>2</td><td>Collision detection (+)</td><td>10</td><td>Transmission (−)</td></tr>
<tr><td>3</td><td>Transmission (+)</td><td>11</td><td>N.C.</td></tr>
<tr><td>4</td><td>N.C.</td><td>12</td><td>Reception (−)</td></tr>
<tr><td>5</td><td>Reception (+)</td><td>13</td><td>+12V</td></tr>
<tr><td>6</td><td>12G</td><td>14</td><td>N.C.</td></tr>
<tr><td>7</td><td>N.C.</td><td>15</td><td>N.C.</td></tr>
<tr><td>8</td><td>N.C.</td><td>Shell</td><td>FG</td></tr>
</table></td></tr>
</table>

(b) Please use cable that meets the transceiver and AUI cable specifications for the transceiver supply power in consideration of the E71 voltage drop (maximum 0.8V).

Remarks

The transceiver power characteristics are

- Input terminal voltage $12V^{-6\%}$ to $15V^{+5\%}$
- AUI cable direct current resistance 40 Ω/km or under, maximum length 50m(164.04 ft.)
- Maximum current consumption 500mA or less

So in consideration of the 0.8V voltage drop of E71 module, the transceiver supply power scale is 14.08V to 15.75V.

* Calculating the voltage drop (V) of the transceiver supply voltage

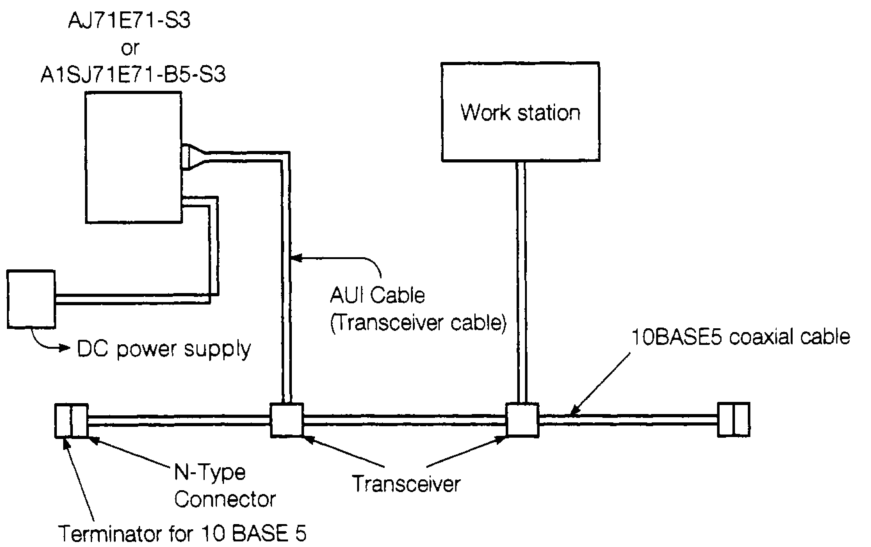
Voltage drop (V) = AUI cable direct current resistance (Ω/m) x AUI cable length (m) x 2 (both directions) x transceiver consumption current (A) + E71 main body voltage drop (V)

(Example)

2.8 (V) = 0.04 (Ω/m) x 50 (m) x 2 x 0.5A + 0.8 (V)

In this case, the target value of the transceiver supply power will be larger than 14.08 V.

14.08 (V) = 12 $V^{-6\%}$ (11.28 V) + 2.8 (V)



* Inquire with a specialized service provider for the devices required.

Fig 2.2 Example Network System Configuration

2 **The equipment required when connecting to 10BASE2 is shown in figure 2.4. The user will please arrange any adapters other than the T-type.**

(a) 10BASE2 coaxial cable

	10BASE2
Transmission VDM	Coaxial cable 50 Ω RG58A/U or RG58C/U

(b) T-type adapter (for connecting to the AJ71E71-S3, A1SJ71E71-B2-S3, included in the packaging)
UG-274/U (HIROSE ELECTRIC) suitable products

(c) BNC plug
BNC-P-58U (DDK ELECTRONICS) or UG-88/U (HIROSE ELECTRIC) suitable products

(d) Terminator for 10BASE2
Plug type terminator type BNC (FUJIKURA LTD.) suitable products

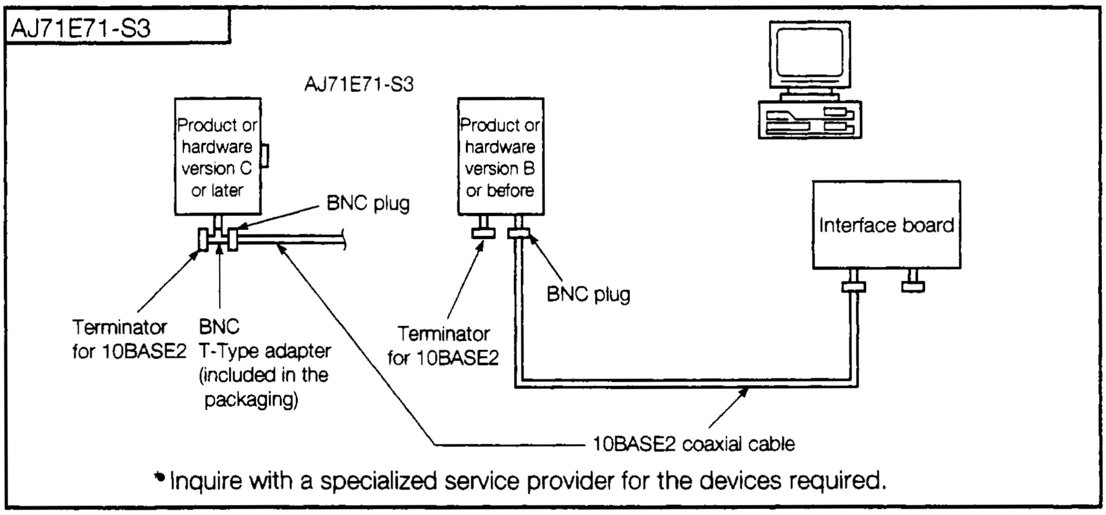


Fig 2.3 Example Network System Configuration

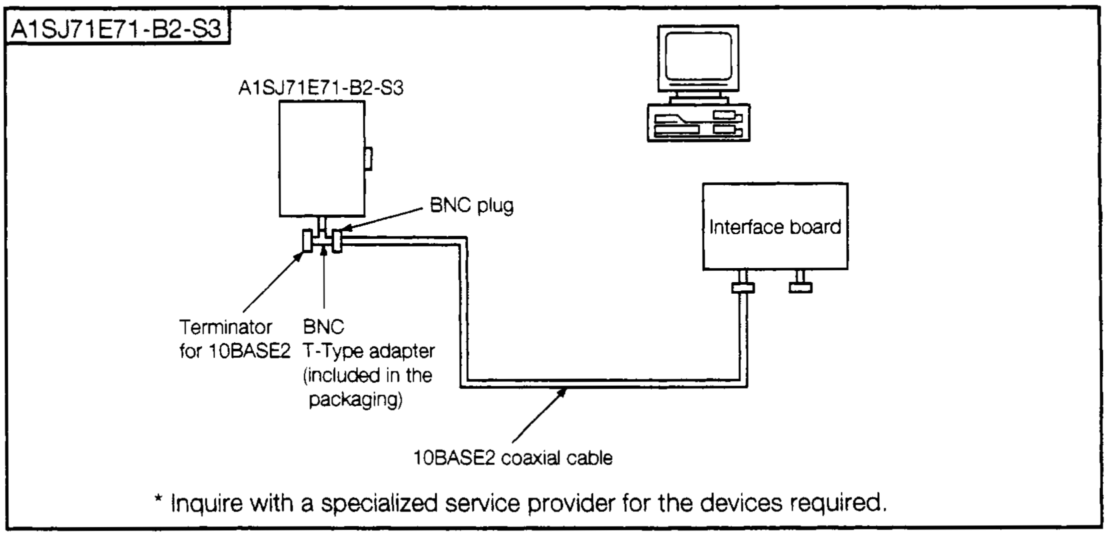


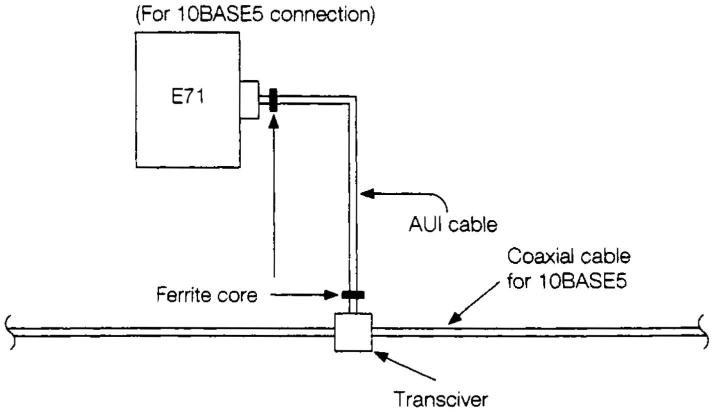
Fig 2.4 Example Network System Configuration

Point

(1) Taking sufficient safety measures is required when installing 10BASE5 and 10BASE2 so please have this work done by a specialist.
(2) To conform to the EMC directives and low voltage directives by incorporating the E71 into a customer's product, install ferrite cores using the method shown in (4) below.
(3) The following countermeasures are available for communication errors caused by the effect of high frequency noise in a given installation environment:
- Install ferrite cores using the method shown in (4) below.
- Increase the communication retry count when TCP/IP communication is performed.

(4) Ferrite cores should be installed as follows for connecting to a network via the 10BASE2 or 10BASE5.
Install a ferrite core (*1) at the E71 side and at the external device side/transceiver side of the AUI cable.
*1: ZCAT 2032-0930 manufactured by TDK can be used.

(For 10BASE5 connection)



MEMO

3. SPECIFICATIONS

This section explains the E71 general specifications, performance specifications, and transmission specifications.

3.1 General Specifications

This section explains the E71 general specifications.

Table 3.1 General Specifications

<table>
<tr><th>Item</th><th colspan="6">Specification</th></tr>
<tr><td>Ambient operating temperature</td><td colspan="6">0 to 55°C</td></tr>
<tr><td>Ambient storage temperature</td><td colspan="6">-20 to 75°C</td></tr>
<tr><td>Ambient operating humidity</td><td colspan="6">10 to 90%RH, Non-condensing</td></tr>
<tr><td>Ambient storage humidity</td><td colspan="6">10 to 90%RH, Non-condensing</td></tr>
<tr><td rowspan="5">Vibration resistance</td><td rowspan="5">Conforming to JIS B3501, IEC 1131-2</td><td></td><td>Frequency</td><td>Acceleration</td><td>Amplitude</td><td>No. of sweeps</td></tr>
<tr><td rowspan="2">Under intermittent vibration</td><td>10 to 57Hz</td><td>—</td><td>0.075mm (0.003inch)</td><td rowspan="4">10 times each in X, Y, Z directions (for 80 min.)</td></tr>
<tr><td>57 to 150Hz</td><td>9.8m/s²</td><td>—</td></tr>
<tr><td rowspan="2">Under continuous vibration</td><td>10 to 57Hz</td><td>—</td><td>0.035mm (0.001inch)</td></tr>
<tr><td>57 to 150Hz</td><td>4.9m/s²</td><td>—</td></tr>
<tr><td>Shock resistance</td><td colspan="6">Conforming to JIS B3501, IEC 1131-2
(147m/s², 3 times in each of 3 directions X Y Z)</td></tr>
<tr><td>Operating ambience</td><td colspan="6">No corrosive gases</td></tr>
<tr><td>Operating elevation</td><td colspan="6">2000m (6562 feet) max.</td></tr>
<tr><td>Installation location</td><td colspan="6">Control panel</td></tr>
<tr><td>Over voltage category *1</td><td colspan="6">II max.</td></tr>
<tr><td>Pollution level *2</td><td colspan="6">2 max.</td></tr>
</table>

*1 This indicates the section of the power supply to which the equipment is assumed to be connected between the public electrical power distribution network and the machinery within the premises. Category II applies to equipment for which electrical power is supplied from fixed facilities. The surge voltage withstand level for up to the rated voltage of 300V is 2500V.

*2 This index indicates the degree to which conductive material is generated in terms of the environment in which the equipment is used. Pollution level 2 is when only non-conductive pollution occurs. A temporary conductivity caused by condensation must be expected occasionally.

3.2 Performance Specifications

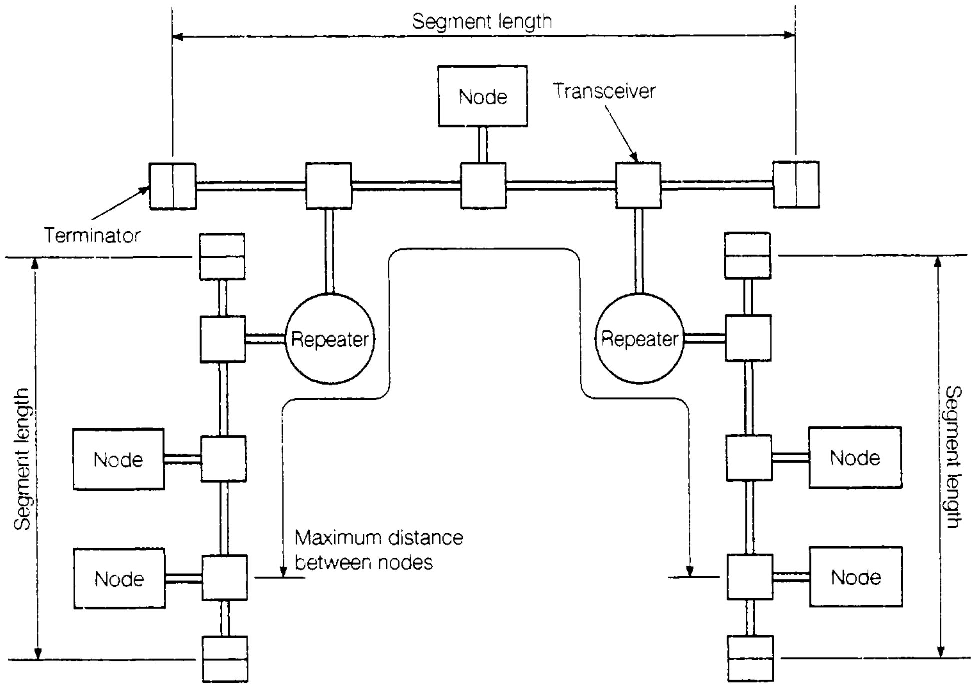
This section explains the E71 performance specifications.

Table 3.2 Performance Specifications

<table>
<tr><th colspan="2" rowspan="2">Item</th><th colspan="2">Specifications</th></tr>
<tr><th>10BASE5</th><th>10BASE2</th></tr>
<tr><td rowspan="5">Transmission specifications</td><td>Data transmission speed</td><td colspan="2">10Mbps</td></tr>
<tr><td>Transmission method</td><td colspan="2">Base Band</td></tr>
<tr><td>Maximum distance between nodes [m(ft.)]</td><td>2500(8202.10)</td><td>925(3034.77)</td></tr>
<tr><td>Maximum segment length [m(ft.)]</td><td>500(1640.42)</td><td>185(606.96)</td></tr>
<tr><td>Maximum number of nodes</td><td>100 Units/Segment</td><td>30 Units/Segment</td></tr>
<tr><td></td><td>Minimum node interval [m(ft.)]</td><td>2.5(8.20)</td><td>0.5(1.64)</td></tr>
<tr><td rowspan="2">Transmission and reception data storage memory</td><td>Fixed buffer</td><td colspan="2">1k Word x 8</td></tr>
<tr><td>Random access buffer</td><td colspan="2">3k Word x 2</td></tr>
<tr><td colspan="2">Number of external nodes that can be communicated with a single initial processing</td><td colspan="2">Max. 20 stations (Refer to Item 5.2 Point)
* More external nodes can be connected by repeating the initial processing.</td></tr>
<tr><td colspan="2">Number of I/O points</td><td colspan="2">32 points</td></tr>
<tr><td colspan="2">5VDC Internal consumption current [A]</td><td colspan="2">AJ71E71-S3 : 1.50 (Hardware version: products before B version)
0.48 (Hardware version: products after C version)···When using 10BASE2
0.26 (Hardware version: products after C version)···When using 10BASE5
A1SJ71E71-B2-S3 : 0.52 (Hardware version: A version)
0.57 (Hardware version: B to D version)
0.49 (Hardware version: products after E version)
A1SJ71E71-B5-S3 : 0.35</td></tr>
<tr><td colspan="2">Noise resistance level</td><td colspan="2" rowspan="3">Power unit specifications for the E71 installed station</td></tr>
<tr><td colspan="2">Voltage resistance</td></tr>
<tr><td colspan="2">Insulation resistance</td></tr>
<tr><td colspan="2">External dimensions [mm(in.)]</td><td colspan="2">AJ71E71-S3 : 250(9.84)x37.5(1.48)x119(4.69)
A1SJ71E71-B2-S3 : 130(5.12)x34.5(1.36)x93.6(3.69)
A1SJ71E71-B5-S3 : 130(5.12)x34.5(1.36)x93.6(3.69)</td></tr>
<tr><td colspan="2">Mass [kg(lb)]</td><td colspan="2">AJ71E71-S3 : 0.60(1.32) (Hardware version: products before B version)
0.52(1.44) (Hardware version: products after C version)
A1SJ71E71-B2-S3 : 0.27(0.59)
A1SJ71E71-B5-S3 : 0.27(0.59)</td></tr>
</table>

Remarks

The maximum distance between nodes and the segment legs are shown in the following diagram.



3.3 Data Codes during Communication and Exchangeable Data Amount

This section explains the data codes used when exchanging between the E71 and remote node or PLC CPU.

1 The data codes used during exchange are given below.

① Between E71 and the remote node

The data exchange function makes it possible to conduct exchange by selecting either binary code or ASCII code as shown in the table below.

Switching between binary code and ASCII code is done using the dip switches (SW2 : Data code setting) on the front of the E71.

(For details refer to Item 4.3.2)

○ : Selection possible
× : Not possible

Table 3.3 Codes that can be Selected

Data Exchange Function		Binary Code	ASCII Code	Function Explanation Item
Fixed buffer exchange	With procedure	○	○	Chapter 6
	Without procedure	○	× (*1)	Chapter 7
Random access buffer exchange		○	○	Chapter 8
Reading/writing data in the PLC CPU (General data exchange)		○	○	Chapter 9 Chapter 10

*1 Communication can be performed using the binary codes shown in the figure below:

② Between E71 and PLC CPU

Communicated binary code.

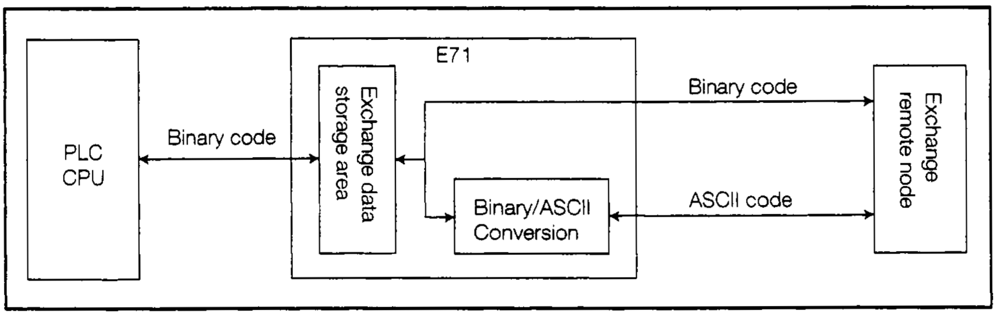


Fig 3.1 Exchange Data Code System

2 **When exchanging using the ASCII code, the 1 byte binary code data is automatically converted into the 2 byte ASCII code.**

Example:

Binary code data		ASCII code data
15_H (One byte)	→	31_H, 35_H "1" "5" (Two bytes)
1234_H (Two bytes)	→	31_H, 32_H, 33_H, 34_H "1" "2" "3" "4" (Four bytes)

3 **The amount of data that can be exchanged at one time between the E71 and an external node depends on the function used and the data code setting (binary/ASCII) selected using the DIP switch (SW2) on the E71 front panel.**

The following table shows the maximum amount of exchange data that can be sent at one time for each data exchange function.

Table 3.4 Exchangeable Data Amount

Data Exchange Function		Binary Code	ASCII Code	Remarks
Fixed buffer exchange	With procedure	1017 Words	1016 Words*1	The same as simultaneous broadcast communication without procedure
	Without procedure	1023 Words	(Cannot be selected)	
Random access buffer exchange		1017 Words	1016 Words*1	
Reading and writing data in the PLC CPU (General data exchange)		The maximum number of operations that can be specified for each command/instruction		Maximum 256 points

*1 Equivalent to 508 words on the PLC CPU.

3.4 Functions

3.4.1 List of Functions

Table 3.5 List of E71 Functions

Functions	Description of Functions		Exchange Partner Unit: Remote Node ↓ E71	E71 ↓ Remote Node	E71 ↓ E71	E71 ↓ AJ71 E71	AJ71 E71 ↓ E71	E71 ↓ QE71	QE71 ↓ E71
Fixed buffer exchange	(1) Exchange between the PLC CPU and remote nodes in the Ethernet is done on a 1:1 basis. When With Procedure is used, exchange is conducted while a handshake is being done with the remote node. (2) When conducting exchange with a remote node, the fixed buffer (one area for each 1k word) has 8 areas (however, please refer to Item 3.3 regarding the amount of data that can be exchanged at one time.) (3) The exchange partner and usage application (transmission/reception) for the fixed buffer is set by the exchange parameters. (4) Exchange can be done with the connection and remote node that is in the Ethernet.	With procedure	○	○	○	○	○	○	○
		Without procedure	○	○	○	○ (*1)	○ (*1)	○	○
Simultaneous broadcast communication	(1) By UDP/IP [fixed buffer exchange without procedure] function, simultaneous broadcast of the appropriate data to all remote nodes within the same Ethernet that E71 is installed can be excuted. However, the remote node must be performed read and delete processing when received message is not required by this simultaneous broadcast communication. (2) Exchange can be done from remote nodes that are ending connection open processing in the Ethernet.		○	○	○	×	×	○	○
Random access buffer exchange	(1) Read and write exchange for the E71 random access buffer memory can be conducted from multiple nodes. (2) Random access buffer used to exchange with remote nodes is 6k words. A continuous area can be read/written from the remote nodes. (However, please refer to Item 3.3 regarding the amount of data that can be exchanged at one time.) (3) During random access buffer exchange, the random access buffer can be used as a common buffer memory within the network without specifying the memory area for each connection. (4) Exchange can be done with remote nodes that are ending the connection open processing in the Ethernet.		○	×	×	×	×	×	×
Reading and writing data in the PLC CPU	(1) The data in the PLC CPU such as that for each device, file data, and special function unit buffer memories, etc., are read and written in the PLC that is installed in the E71 using request from the remote node. (2) When the PLC installed in the E71 is connected to the MELSECNET, the exchange from remote node with the remote station PLC CPU can be done via the MELSECNET. (Please refer to Item 9.1 for details regarding data exchange with remote stations.) (3) Exchange can be done with remote node that is doing connection open processing end in the Ethernet. In addition, if the data exchange function is used, exchange can be done even if the local PLC CPU is stopped.		○	×	×	×	×	×	×

Functions	Description of Functions	Exchange Partner Unit						
		Remote Node ↓ E71	E71 ↓ Remote Node	E71 ↓ E71	E71 ↓ AJ71 E71	AJ71 E71 ↓ E71	E71 ↓ QE71	QE71 ↓ E71
Router relay exchange	(1) Exchanges data via the routers that are connected in the Ethernet network system. (Do not operate as a router.) (2) Exchange is possible via a router by data transmission after TCP's active open and UDP open. (3) Exchange can be done with the remote node that is ending the connection open processing in the Ethernet.	○	○	○	○	×	○	○
Existence check	(1) Checks if the partner node is operating correctly when exchange has not been done with the partner node for a specified period of time after connection open processing has ended. (2) Closes the line (connection forced disconnect) if the E71 DIP switch (SW1) is off when the partner node is not operating correctly.	○	○	○	○	×	○	○
Exchanged error storage	(1) Stores a maximum of 10 sets of error history information, such as message subheaders and partner node IP address, in the buffer memory when a data exchange error occurs. (2) This error history infoemation makes it easy to analyze the cause of data exchange troble.							
Self wrap-ping test	(1) Conducts a hardware check including E71 transmission and reception circuits.							

○ : Executable × : Not executable

*1 Procedures must be created using the sequence program.

3.4.2 Relationship between Communication Remote Node and Added Functions for Each Communication Function

Shows what partner equipment can be exchanged with and what added functions can be used for each exchange function.

Table 3.6 Relation between Exchange Partner Nodes and Added Functions

Exchange Functions		Communication Format with Exchange Partner Nodes		Added Functions			
		TCP/IP	UDP/IP	Simultaneous Broadcast	Pairing Exchange (*1)	Existence check	Exchange when PLC CPU is stopped
Fixed buffer exchange	With procedure	○	○	×	○	○	×
	Without procedure	○	○	○ (*2)	○	○	×
Random access buffer exchange		○	○	×	—	○	○
Reading and writing Data in the PLC CPU		○	○	×	—	○ (*3)	○
Router relay exchange (router relay functions)		○	○	×	—	—	—

*1 For information regarding pairing exchange please refer to Item 5.4.1 1 (b) ③ and Item 5.4.4.

*2 Only valid during UDPs without procedure exchange.

*3 Only valid with nodes opened by the sequence program.

3.5 Send and Receive Processing

3.5.1 Message Division and Data Length

The message exchanged between the E71 and the remote node is sometimes allocated by the local node or remote node transmission/reception buffer capacity.

Below is shown the message data length when it is allocated and transmitted or received.

1 **When the E71 receives data the message is reconstructed using the following data link (hereafter internal data length) when the message is reconstructed from the TCP/UDP level.**

① Data length in the data bearing fixed buffer exchange or random access buffer exchange.

② The data length calculated from the command message contents when data is read or written in the PLC CPU.

2 **The E71 performs the following processing when the internal data length and the data amount that is actually transmitted differ.**

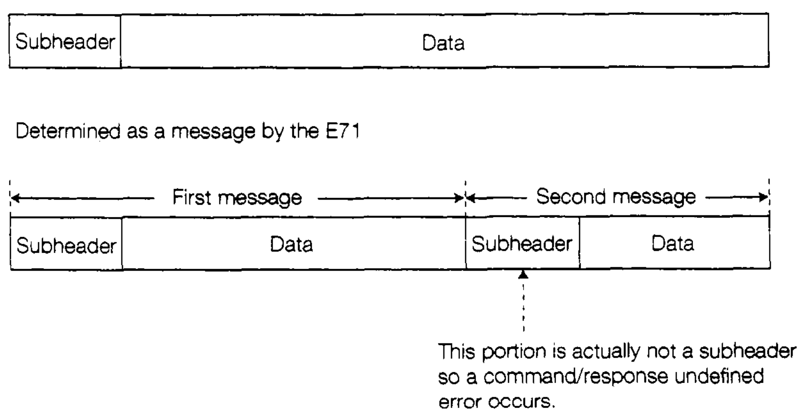
① When the transmitted amount is less

- Waits until the remaining data is transmitted.
- If the next data is not sent within the response monitoring time the message currently being received will be ignored and the next processing (receive rate form initial message) will begin.

② When the transmitted data amount is more

- Determined to be a transmission with two or more messages linked together.
- The data from the first data received until the internal data length is handled as the initial message and that coming immediately thereafter is handled as the next message. (An error will be generated if there is subheader analysis being conducted or if processing is being conducted for the second message or thereafter.)

(Example) Message transmitted has one message from the partner node



Remarks

(1) If the error mentioned in 2 above is generated, an error code is stored for one of the error information storage areas shown below in the E71 buffer memory.

- Exchange state storage area
- Error log area

3.5.2 Continuous Processing Over the Same Connection

The following explains the procedure to transmit data between E71 and other nodes over the same connection, and the E71 processing when continuous data transmission is performed from other nodes to the E71 over the same connection.

1 **Procedure to transmit data between other nodes and E71**

When transmitting data between other nodes and E71, perform data transmission after the processing for the previous data transmission is complete.

For example, when data transmissions accompany responses, the next data transmission should be commenced after the response to the previous transmission is received.

* If data transmission is continued without following the above procedure, an error may be generated on the E71 side, the connection may be closed, or the open error detection signal (X18) may turn on.

2 **The E71 processing during continuous data transmission is performed from other nodes to E71**

The following explains the E71 processing when it receives a new message that has no relationship to the current processing, which is the processing of the message received previously over the same connection.

(a) The processing of the message received first is performed.

(b) After the processing of the message received first is complete, the processing of the next message will be commenced. (The processing of the next message will be suspended.)

Example: When a random access buffer read request is received via connection 1 while the receive processing for fixed buffer exchange is in progress

① The receive processing for the fixed buffer exchange is continued (*1).

② After the receive processing for the fixed buffer exchange is completed, the processing for random access buffer read is executed.

*1 The "receive processing for fixed buffer exchange in progress" is defined as the period of time from when the receive end signal (X0) turns on until the receive end confirmation signal (Y0) turns off.

3.5.3 Conditions for Issuing a Forced Disconnect

When the following conditions occur the E71 sends a connection forced disconnect to the remote node to forcefully close the line. (ABORT (RST) command transmission)

① When the initial request signal is turned off in the open state.

② When the remaining messages are not received within the response monitoring time when data allocation is being received.

③ A TCP transmission error occurs when line close when TCP ULP timie out error occurs is selected using the DIP switch (SW1).

④ When an existence check trouble is detected for the existence check connection.

⑤ When either active or full-passive open for performing TCP/IP communication has normally been completed, an open request is received again from the remote node side.
Or when an open request with the same port number is received again from the identical remote node while non-passive open has normally been ended.

3.6 I/O Signals for the PLC CPU

This section explains the E71 I/O signals.

The I/O signal allocation shows that the E71 is installed in the 0 slot of the basic unit. It means that device X receives an input signal from the E71 to the PLC CPU and that device Y receives an output signal from the PLC CPU to the E71.

3.6.1 List of I/O Signals

Table 3.7 List of I/O Signals for the PLC CPU

Signal Direction E71 to PLC CPU			Signal Direction PLC CPU to E71		
Device No.	Signal Name		Device No.	Signal Name	
X0	Transmission normal end or reception end signal	For connection 1's fixed buffer exchange	Y0	Connection number 1	Transmission request or reception end check signal
X1	Transmission error detection signal or reception error detection signal		Y1	Connection number 2	
X2	Transmission normal end or reception end signal	For connection 2's fixed buffer exchange	Y2	Connection number 3	
X3	Transmission error detection signal or reception error detection signal		Y3	Connection number 4	
X4	Transmission normal end or reception end signal	For connection 3's fixed buffer exchange	Y4	Connection number 5	
X5	Transmission error detection signal or reception error detection signal		Y5	Connection number 6	
X6	Transmission normal end or reception end signal	For Connection 4's fixed buffer exchange	Y6	Connection number 7	
X7	Transmission error detection signal or reception error detection signal		Y7	Connection number 8	
X8	Transmission normal end or reception end signal	For connection 5's fixed buffer exchange	Y8	Connection number 1	Open request signal
X9	Transmission error detection signal or reception error detection signal		Y9	Connection number 2	
XA	Transmission normal end or reception end signal	For connection 6's fixed buffer exchange	YA	Connection number 3	
XB	Transmission error detection signal or reception error detection signal		YB	Connection number 4	
XC	Transmission normal end or reception end signal	For Connection 7's fixed Buffer Exchange	YC	Connection number 5	
XD	Transmission error detection signal or reception error detection signal		YD	Connection number 6	
XE	Transmission normal end or reception end signal	For connection 8's fixed buffer exchange	YE	Connection number 7	
XF	Transmission error detection signal or reception error detection signal		YF	Connection number 8	
X10	For connection number 1	Open end signal	Y10	Usage prohibited	
X11	For connection number 2		Y11		
X12	For connection number 3		Y12		
X13	For connection number 4		Y13		
X14	For connection number 5		Y14		
X15	For connection number 6		Y15		
X16	For connection number 7		Y16		
X17	For connection number 8		Y17	COM.ERR LED turn off request signal	
X18	Open error detection signal		Y18	Usage prohibited	
X19	Initial normal end signal		Y19	Initial request signal	
X1A	Initial error detection signal		Y1A	Usage prohibited	
X1B	Usage prohibited		Y1B		
X1C	COM.ERR LED turned on signal		Y1C	Buffer memory channel switching	
X1D	Usage prohibited		Y1D	Usage prohibited	
X1E			Y1E		
X1F	Watchdog timer error detection signal		Y1F		

Important

Of the output signals directed to the special function module from the PLC CPU, do not output (switch on) the signals that are "use-prohibited".

If output is performed with respect to a "use-prohibited" signal, it may result in the malfunctioning of the PLC system.

3.6.2 Detailed Explanation of I/O Signals

This section explains about the I/O signals ON/OFF timing and conditions shown in table 3.7 (on the previous page). The codes in the parentheses are the device numbers that correspond to table 3.7.

1 Transmission normal end and reception normal end (X0, X2, X4, X6, X8, XA, XC, XE)

These signals are used when exchange is conducted with the fixed buffer. These signals are not used when exchanging with the random access buffer or when reading and writing data in the PLC CPU. Used as the transmit normal end signal when the appropriate fixed buffer transmission is used. Used as the reception end signal when the appropriate fixed buffer reception is used.

(a) When used as transmission end signal

When exchanging with procedure

① Data is transmitted when the data request signal (Y0 to Y7) is on.

② The remote node that has received the data returns a response to the E71.

③ The transmit normal end signal is turned on when a response is returned by the remote node.

④ The transmit normal end signal is turned off when the transmit request signal (Y0 to Y7) is turned off.

⑤ The transmission normal end signal is not turned on when the end code returned from the remote node is anything other than 00H. The transmission error detection signal (X1, X3, X5, X7, X9, XB, XD, XF) is turned off.

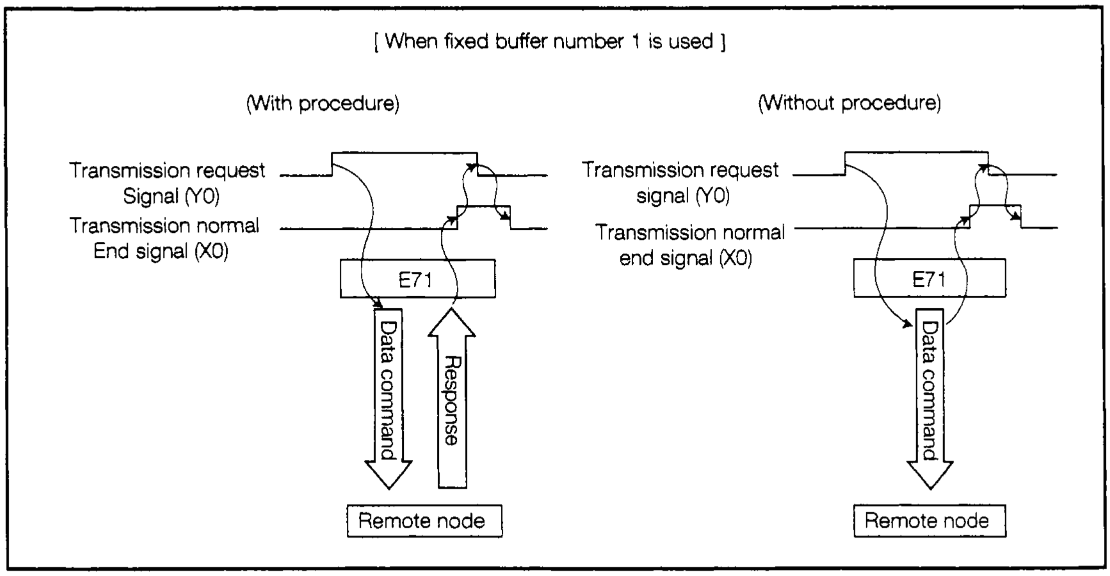
When exchanging without procedure

① The data is transmitted when the transmission request signal (Y0 to Y7) is turned on.

② The transmission normal end signal is turned on when the data transmission is completed.

③ The transmission normal end signal is turned off when the transmission request signal (Y0 to Y7) is turned off.

④ The transmission normal end signal is not turned on when a transmission error is generated. The transmission error detection signal (X1, X3, X5, X7, X9, XB, XD, XF) is turned on.



(b) When used as reception end signal

When exchanging with procedure

① Turns on when the E71 receives data from a remote node.

② Can be used as the handshake signal when reception data is read from the PLC CPU using the FROM command, etc.

③ The reception end check signal (Y0 to Y7) is turned on after the reception data is read using the FROM command, etc.

A response is returned to the remote node that transmitted the data.

④ Reception end signal is automatically turned off after a response is sent to the remote node.

⑤ The reception end signal is not turned on when problem data is sent from the remote node.

When exchanging without procedure

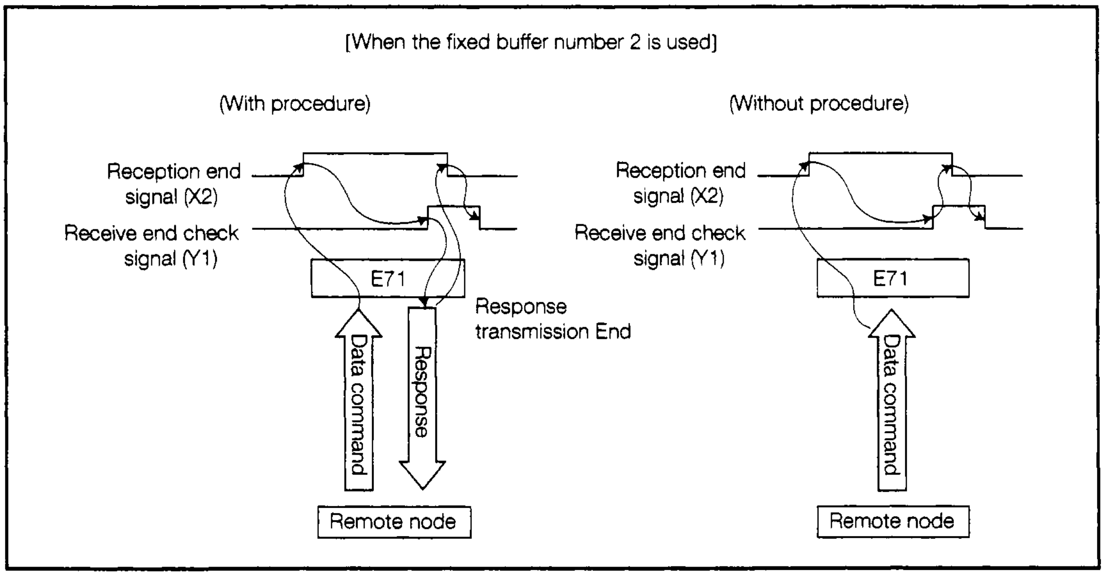
① Turns on when the E71 receives data from a remote node.

② Can be used as the handshake signal when the received data is read from the PLC CPU using a FROM command, etc.

③ The receive end check signal (Y0 to Y7) is turned on after the receive data is read using the FROM command, etc.

④ The receive end signal is automatically turned off when the receive end check signal is turned on.

⑤ The receive end signal is not turned on when following data is transmitted from a remote node.



2 Transmission error detection signal or reception error detection signal (X1, X3, X5, X7, X9, XB, XD, XF)

This signal is used when fixed buffer exchange is performed. This signal is not used for random access buffer exchange or exchange when reading or writing of data inside the PLC CPU is conducted.

When the correspongind fixed buffer is used for transmission, use it as the transmission error detection signal.
When the corresponding fixed buffer is used for reception, use it as the reception error detection signal.

(a) When used as the transmission error detection signal

When exchanging with procedure

① The transmit error detection signal is turned on when the response from the remote node is not returned within the response monitor time (Refer to Item 5.3.1) after data is transmitted from the fixed buffer.

② The transmission error detection signal is turned on when the specified retry processing (Refer to Item 5.3.1) is conducted when the "ACK" is not returned after data is sent from the fixed buffer that is using the TCP connection. (Retry processing is not performed for UDP)

③ The transmission error detection signal is turned on when the finish code response received from the remote node is anything other than 00H after data has been transmitted from the fixed buffer.

④ When the transmission error detection signal is turned on, the error contents can be checked by reading the fixed buffer transmission error code storage area (buffer memory 94, 104, ...164).

⑤ The transmission error detection signal is turned off when the fixed buffer transmission request signal (Y0 to Y7) is turned off.

When exchanging without procedure

① The transmission error detection signal is turned on when the specified retry processing (Refer to Item 5.3.1) is performed when the "ACK" is not returned after the data has been transmitted from the fixed buffer using the TCP connection (Retry processing is not performed for UDP.)

② When the transmission error detection signal is turned on, the error contents can be checked by reading the fixed buffer transmission error code storage area (buffer memory 94, 104, ...164).

③ The transmission error detection signal is turned off when the fixed buffer transmission request signal (Y0 to Y7) is turned off.

(b) When used as the reception error detection signal

① When close processing is performed for the corresponding connection upon receiving Close/Abort (RST) from the partner remote node before the completion of the previous data reception processing, the reception error detection signal may turn on.

② When Close/Abort (RST) is received from the partner remote node after the completion of data reception processing, the reception error detection signal does not turn on.

③ If the reception error detection signal is on when close processing is performed for the corresponding connection according to (1) above, turn off the open request signal (Y8 to YF) after turning off the reception completion confirmation signal at the timing shown in Remarks in 2 of Item 5.4.3.

3 **Open end signal (X10 to X17)**

(a) When the connection open request signal (Y8 to YF) is turned on by the sequence program the exchange parameters are checked and open processing is executed. Here if open processing is performed normally the open end signal (X10 to X17) is turned on.

(b) When the open request signal is turned on and open processing is not performed normally the open error detection signal (X18) is turned on. (In this case the open end signal is not turned on.)

(c) Only data can be exchanged with the remote node for which the open end signal (X10 to X17) is turned on (fixed buffer exchange, random access buffer exchange, and reading and writing from the sequence CPU exchange).

(d) The open end signal (X10 to X17) ON/OFF can be checked using the LED display (BUF1 to BUF8) on the front of the E71.

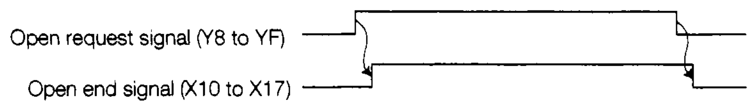
(e) The open end signal (X10 to X17) is turned off when the open request signal is turned off by the sequence program. In the following case the open end signal (X10 to X17) is not turned off. (Refer to Item 5.4.3)

① When an error is generated.

② When CLOSE or ABORT commands are received from the exchange remote node.

③ When a response monitor timer error occurs.

④ When an error occurs with an existence check function.



4 **Open error detection signal (X18)**

(a) When the connection open request signal (Y8 to YF) is turned on by the sequence program the exchange parameters are checked and if an error is detected the open error detection signal is turned on.

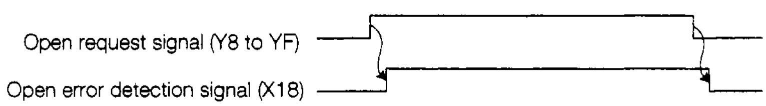
(b) The open error detection signal is turned on when the open request signal (Y8 to YF) is turned on and open processing is not performed normally.

(c) The open error detection signal is turned on when the TCP or ULP time-out error is generated when the E71 main unit's DIP switch SW1 (line processing selection during TCP time-out error) is turned off.

(d) When the open error detection signal is on the error contents of the connection for which an error has occurred can be checked by reading the error codes such as open error code storage area (buffer memory 93, 103, ...163) or the error log lock in the exchange state storage area.

(e) The open error detection signal (X18) is turned off when the open request signal (Y9 to YF) is turned off for the connection for which the open error is occurring.

(f) When there are multiple open errors the open error detection signal (X18) will not turn off unless all of the corresponding open request signals are turned off.



5 **Initial normal end signal (X19)**

(a) When the initial request signal (Y19) is turned on by the sequence program the initial parameters are checked and initial processing is executed. Here if initial processing is conducted normally the initial normal end signal (X19) is turned on. (The module's RDY LED will flash after normal end.)

(b) If the initial processing is not conducted normally the initial error detection signal (X1A) is turned on. (In this case the initial normal end signal (X19) is not turned on.)

6 **Initial error detection signal (X1A)**

(a) The initial error detection signal (X1A) is turned on and the initial request signal (Y19) is on and initial processing does not end normally.

(b) The initial error detection signal turns on when a hardware error occurs after initial processing ends normally. (The initial normal end signal is turned off.)

(c) When the initial error detection signal (X1A) is turned on, the error contents can be checked by reading the exchange state storage area's initial error code storage area (Refer to Item 5.5.1 buffer memory 80).

(d) The initial error detection signal (X1A) is turned off when the initial request signal is turned off.

7 **COM.ERR LED tumed on signal (X1C)**

(a) The COM.ERR LED turned on signal (X1C) is turned on when the COM.ERR LED is turned on when and exchange error occurs. (Refer to Item 13.2*1)

(b) The COM.ERR LED turned on signal (X1C) is turned off when the COM.ERR LED turn off request signal (Y17) of the sequence program is turned on.

8 **Watchdog timer error detection signal (X1F)**

The watchdog timer detection signal (X1F) is turned on when a watchdog timer (approximately 300 ms) error occurs when the E71 self diagnostic is used.

9 **Transmission request and reception end check signal (Y0 to Y7)**

This signal is used when fixed buffer exchange is conducted.

This signal is not used for random access buffer exchange and reading and writing data from the PLC CPU exchange.

Used as a transmission request signal when the appropriate fixed buffer is used for transmission.

Used as the reception end check signal when the appropriate fixed buffer is used for reception.

(a) When used as transmission request signal

When exchanging with procedure

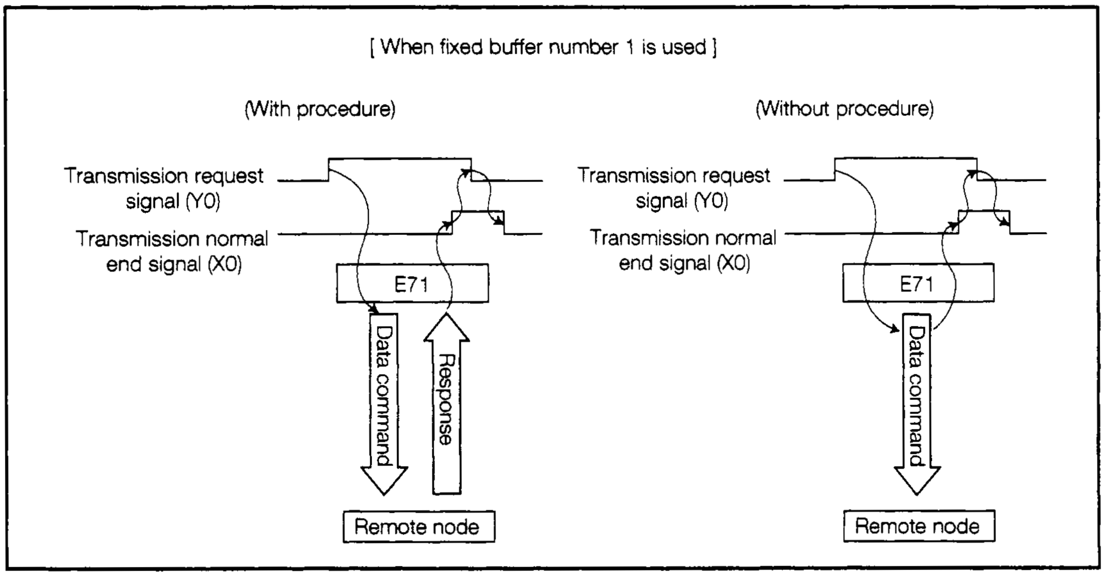
① The E71 transmits data to the node specified by the parameters when the transmission request signal (Y0 to Y7) is turned on by the sequence program.

② Transmission is ended when the transmission end signal (X0: when the fixed buffer is No. 1) when a response is returned from the remote node after data transmission.

When exchanging without procedure

① Data is transmitted by the E71 to the remote node specified by the parameters when the transmission request signal (Y0 to Y7) is turned on by the sequence program.

② Transmission ends when the transmission end signal (X0: when the fixed buffer is No. 1) is turned on after the data is transmitted.



(b) When used as reception end check signal

When exchanging with procedure

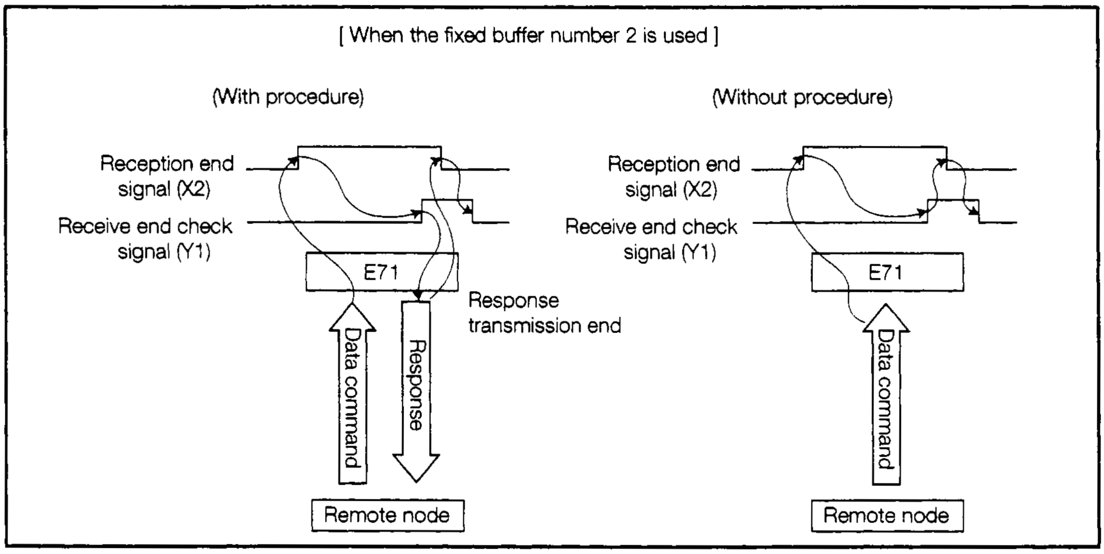
① The reception end signal (X2 : when the fixed buffer is number 2) is turned on after the E71 receives data from the remote node.

② A response is returned to the remote node when the reception end check signal (Y0 to Y7) is turned on after the reception end signal (X2: when the fixed buffer is number 2) is in the on state after check by the sequence program.

When exchanging without procedure

① The reception end signal (X2 : when the fixed buffer is No. 2) is turned on after data is received to the E71 from the remote node.

② The reception end signal is turned off when the reception end check signal (Y0 to Y7) is turned on and the reception end signal (X2: when the fixed buffer is No. 2) is in the on state after check by the sequence program.



10 Open request signal (Y8 to Y7)

(a) This is the on signal for exchanging data (fixed buffer exchange, random access buffer exchange, reading and writing from the PLC CPU exchange) between the E71 and the remote node.

(b) When each connection open request signal (Y8 to YF) is turned on by the sequence program the exchange parameters are checked and if normal open processing is conducted. If error is detected the open error detection signal (X18) is turned on.

(c) When the open request signal is on and open processing is conducted normally the open end signal (X10 to X17) is turned on. If an error is detected the open error detection signal (X18) is turned on.

(d) The open error detection signal (X18) is turned off when the open request signal (Y8 to YF) is turned off. When errors occur at multiple connections, the open request signal for all connections where errors are occurring are turned off.

When the open error detection signal (X18) is turned ON, be sure to read the open error code storage area (buffer memory 93, 103, ...163) before the open request signal is turned ON again.

(e) When turning OFF the open request signal, be sure that the I/O signals and other items related to the corresponding connection to the E71 are in the following status:
- Transmission request signal/receive end check signal (Y0 to Y7), transmission normal end signal/reception end signal (X0, X2 and beyond), transmission error detection signal (X1, X3 and beyond) are all OFF.
- Open end signal (X10 to X17) is ON. Or, if the open error detection signal (X18) is ON, the open error code for the corresponding connection is other than 0.

* When "passive open" is specified in the open processing of TCP/IP communication, the open request signal can be turned OFF before the open end signal or open error detection signal turns ON. In such cases, when reopening processing using passive open, turn ON the open request signal (OFF → ON) once 500 ms have passed after the open request signal is turned OFF.

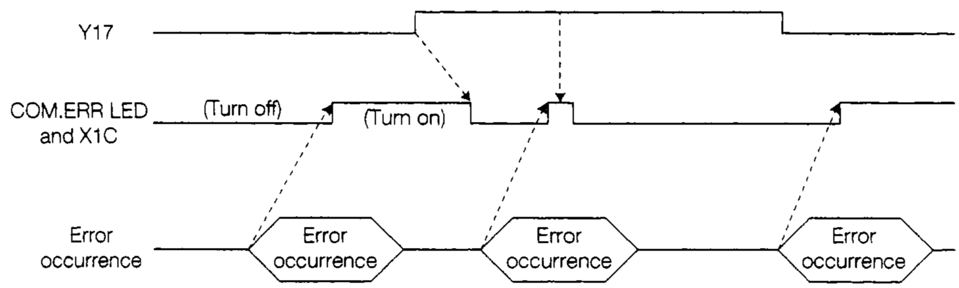
11 COM.ERR LED turn off request signal (Y17)

This signal is used to turn off the COM.ERR LED on the front of the E71 that is lit when an exchange error occurs.

(a) The COM.ERR LED is turned off when the turn off request signal (Y17) is turned on.

(b) Normally, turn off processing is conducted when the turn off request signal (Y17) is on.

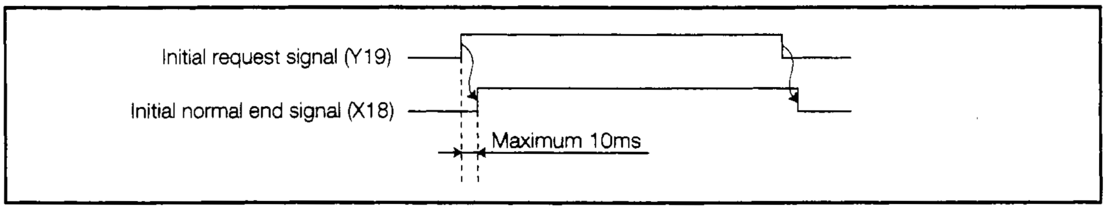
(c) When the turn off request signal (Y17) is turned on, the error information in the buffer memory error log area is cleared (deleted).



12 Initial request signal (Y19)

(a) This signal is for conducting initialization before the E71 conducts exchange.

(b) Turning on the initial request signal (Y19), checks the initial parameters and if they are normal executes initial processing. If an error is detected the initial error detection signal (X1A) is turned on.



(c) When the initial request signal is turned on and initial processing is conducted normally, the initial normal end signal (X19) is turned on. Here, if an error is detected, the initial error detection signal (X1A) is turned on.

(d) When an E71 system error occurs the initial error detection signal (X1A) is turned on. Please turn off the initial request signal (Y19). The error code at this time is stored in the initial error code storage area (buffer memory 80).

(e) After checking that the following signals are turned off, turn off the initial request signal.

- Transmission request signal/reception end check signal (Y0 to Y7)
  Transmission normal end signal/reception end signal (X0, X2...)
  Transmission error detection signal (X1, X3...)
- Open request signal (Y8 to YF)
  Open end signal (X10 to X17)
  Open error detection signal (X18)

13 **Buffer memory channel switching signal (Y1C)**

This signal is used to specify the buffer memory channel. It is turned on/off by the sequence program before the read/write to the E71 buffer memory (*1) is conducted by the sequence program's FROM/TO instruction.

OFF : Channel 0 becomes valid.

ON : Channel 1 becomes valid.

*1 When the PLC CPU conducts data read/write for the fixed buffer (address: 512 to 4607) and the random access buffer (address: 4608 to 7679), the buffer memory channel switching signal (Y1C) is turned on/off by the PLC CPU. Turning the buffer memory channel switching signal (Y1C) on/off conducts data read/write to the next area.

- When the buffer memory channel switching signal (Y1C) is off

  Fixed buffer No. 1 to No. 4 area, random access buffer (first half 3k words)

- When the buffer memory channel switching signal (Y1C) is on

  Fixed buffer No. 5 to No. 8 areas, random access buffer (last half 3k words)

When the I/O control method of the PLC CPU installed in the E71 is the refresh method, one of the following is performed when the above buffer memory read/write is conducted after the buffer memory channel switching signal (Y1C) is turned from on to off/off to on.

① The next sequence scan is read/written after the buffer memory channel switching (Y1C) is turned on/off.

② Read/write is performed after the signal (Y1C) range is refreshed by the common instruction's SEG after the buffer memory channel switching signal (Y1C) is turned on/off.

③ Read/write is performed after the buffer memory channel switching signal (Y1C) is direct set/reset by the dedicated instruction's DSET/DRST.

④ Read/write is conducted after the signal (Y1C) is direct output by the dedicated instruction's DOUT after the buffer memory channel switching signal (Y1C) is turned on/off.

3.7 Buffer Memory

This section explains about the PLC CPU and the data reception buffer memory in the E71.

3.7.1 Buffer Memory Applications

The buffer memory is composed of the following user areas and system area.

1 User area

① This is the area other than the system areas given below.

② These areas are the areas where the parameter types are set for initial processing and data exchange, data exchange areas, and areas for storing exchange state and exchange error information.

③ The default values used when the E71 is booted up are stored in the parameter settings area for initial processing and data exchange. These default values make exchange with a remote node possible, but sometimes they need to be changed. Before conducting processing, write the set values only into the areas that need to be changed.

④ When reading from and writing to the user area, please follow the detailed instruction items. In addition, only execute post reading and writing (FROMP, TOP, etc.) when necessary. The data exchange time sometimes becomes long during normal execution.

2 System area

This is the area used by the E71.

In addition to the system areas shown in Item 3.7.2, system areas partially exist in the user areas described in 1 above.

Important

Do not write data in the "system area" in the buffer memory of the special function module.
If data is written to the "system area", it may result in the malfunctioning of the PLC system.

Point

The access from PLC CPU is given priority over other special function module processing.
Therefore, if the PLC CPU frequently accesses the buffer memory of a special function module, not only the scan time of the PLC CPU is prolonged, but each processing of the special function module is delayed.
Perform access to the buffer memory from the PLC CPU using FROM/TO instruction or other means only when it is necessary.

3.7.2 List of Buffer Memory Allocations

The buffer memory is comprised of 1 address of 16 bits. The overall configuration of the buffer memory is shown below. For details regarding each area, please refer to the explanation references shown on the right side of each area. (How to read the buffer memory address)

Addresses are written using the following method and the same explanation is given hereafter.

□ H (□)

- (□) — The address for this area is shown in decimal numbers.
- □ H — The address for this area is shown in hexadecimal numbers.

(address)	Buffer Memory		Detailed Explanation
0 to FH (0 to 15)	Initial processing parameter settings area (16 words)		Item 5.3.1
10 to 4FH (16 to 79)	Exchange parameter setting area (64 words)		Item 5.4.1
50 to 55H (80 to 85)	Exchange state storage area	Initial processing state storage area (6 words)	Item 5.5.1
56 to 58H (86 to 88)		System area (Use prohibited...3 words)	————
59 to A8H (89 to 168)		Exchange state storage area (80 words)	Item 5.5.2
A9 to B3H (169 to 179)		Error log area (11 words)	Item 5.5.3
B4 to 16FH (180 to 367)		System area (Use prohibited...188 words)	————
170 to 1BFH (368 to 447)		Each protocol state storage area (80 words)	Item 5.5.4
1C0 to 1C1H (448 to 449)	Sub-net mask settings area (2 words)		Item 11.2
1C2 to 1D8H (450 to 472)	Routing information area (23 words)		Item 12.4
1D9 to 1EFH (473 to 495)	Systems area (Use prohibited...23 words)		————
1F0H (496)	Instructions area	Exchange specification during STOP area(1 Word)	Item 5.4.1
1F1 to 1FFH (497 to 511)	Systems area (Use prohibited...15 words)		————
200 to 5FFH (512 to 1535)	Fixed buffer No. 1 (1024 words)	Fixed buffer No. 5 (1024 words)	Chapter 6 Chapter 7
600 to 9FFH (1536 to 2559)	Fixed buffer No. 2 (1024 words)	Fixed buffer No. 6 (1024 words)	
A00 to DFFH (2560 to 3583)	Fixed buffer No. 3 (1024 words)	Fixed buffer No. 7 (1024 words)	
E00 to 11FFH (3584 to 4607)	Fixed buffer No. 4 (1024 words)	Fixed buffer No. 8 (1024 words)	
1200 to 1DFFH (4608 to 7679)	Random access buffer (First half 3072 words)	Random access buffer (Last half 3072 words)	Chapter 8

Access to the fixed buffer and random access buffer by the PLC CPU.

- Access is possible with Y1C = OFF. (Channel 0 specified)
- Access is possible with Y1C = ON. (Channel 1 specified)

(Address)	Buffer Memory		Default Values
0 to 1H (0 to 1)	Local station E71's IP address	(2 words)	0H (0)
2H (2)	Special function settings	(1 word)	0H (0)
3H (3)	Timer setting time units	(1 word)	7D0H(2000)
4 to 6H (4 to 6)	System area (Use prohibited)	(3 words)	—
7H (7)	Destination existence check start interval timer value	(1 word)	12CH (300)
8H (8)	Destination existence check interval timer value	(1 word)	5H (5)
9H (9)	Number of retransmit tries for destination existence check	(1 word)	3H (3)
AH (10)	TCP/ULP time out value	(1 word)	FH (15)
BH (11)	TCP zero window timer value	(1 word)	5H (5)
CH (12)	TCP retransmit timer value	(1 word)	5H (5)
DH (13)	TCP end timer value	(1 word)	AH (10)
EH (14)	IP setup timer value	(1 word)	3H (3)
FH (15)	Response monitoring timer value	(1 word)	FH (15)
10H (16)	Connection No.1	Usage available settings area (1 word each)	0H (0)
11H (17)	Connection No.2		0H (0)
12H (18)	Connection No.3		0H (0)
13H (19)	Connection No.4		0H (0)
14H (20)	Connection No.5		0H (0)
15H (21)	Connection No.6		0H (0)
16H (22)	Connection No.7		0H (0)
17H (23)	Connection No.8		0H (0)
18H (24)	E71's Port No.	Exchange address settings area (For Connection No.1 7 words)	0H (0)
19H (25) 1AH (26)	Remote node IP address		0H (0)
1BH (27)	Remote node port No.		0H (0)
1CH (28) 1DH (29) 1EH (30)	Remote node (L) Ethernet to Address (H)		FFFFFFFFFFFFH
1F to 25H (31 to 37)	E71's port No. to	Exchange address settings area (For connection No.2 7 words)	(Same as connection No.1)
26 to 2CH (38 to 44)	E71's port No. to	Exchange address settings area (For connection No.3 7 words)	(Same as connection No.1)
2D to 33H (45 to 51)	E71's port No. to	Exchange address settings area (For connections No.4 7 words)	(Same as connection No.1)
34 to 3AH (52 to 58)	E71's port No. to	Exchange address settings area (For connections No.5 7 words)	(Same as connection No.1)
3B to 41H (59 to 65)	E71's port No. to	Exchange address settings area (For connections No.6 7 words)	(Same as connection No.1)
42 to 48H (66 to 72)	E71's port No. to	Exchange address settings area (For connections No.7 7 words)	(Same as connection No.1)
49H (73)	E71's port No.	Exchange address settings area (For connections No.8 7 words)	0H (0)
4AH (74) 4BH (75)	Remote node IP address		0H (0)
4CH (76)	Remote node port No.		0H (0)
4DH (77) 4EH (78) 4FH (79)	Remote node (L) Ethernet to Address (H)		FFFFFFFFFFFFH

(Address)	Buffer Memory			Default Value
50H (80)	Initial error code		(1 word)	0H (0)
51 to 52H (81 to 82)	Local station E71's IP address		(2 words)	0H (0)
53 to 55H (83 to 85)	Local station E71's Ethernet address		(3 words)	0H (0)
56 to 58H (86 to 88)	System area (Use prohibited)		(3 words)	———
59H (89)	Local station E71's port No.		Information by Connection (10 words for connection No.1)	0H (0)
5A to 5BH (90 to 91)	Remote node IP address			0H (0)
5CH (92)	Remote node port No.			0H (0)
5DH (93)	Open error code			0H (0)
5EH (94)	Fixed buffer transmission/reception error code			0H (0)
5FH (95)	Fixed buffer exchange end code			0H (0)
60H (96)	Maximum value	Fixed buffer exchange's exchange time		0H (0)
61H (97)	Minimum value			0H (0)
62H (98)	Current value			0H (0)
63 to 6CH (99 to 108)	Local station E71's port No. to		Information by connection (For connection No.2)	(Same as above)
6D to 76H (109 to 118)	Local station E71's port No. to		Information by connection (For connection No.3)	(Same as above)
77 to 80H (119 to 128)	Local station E71's port No. to		Information by connection (For connection No.4)	(Same as above)
81 to 8AH (129 to 138)	Local station E71's port No. to		Information by connection (For connection No.5)	(Same as above)
8B to 94H (139 to 148)	Local station E71's port No. to		Information by connection (For connection No.6)	(Same as above)
95 to 9EH (149 to 158)	Local station E71's port No. to		Information by connection (For connection No.7)	(Same as above)
9F to A8H (159 to 168)	Local station E71's port No. to		Information by connection (For connection No.8)	(Same as above)
A9H (169)	Error log	Area-1	(1 word)	0H (0)
AAH (170)		Area-2	(1 word)	0H (0)
ABH (171)		Area-3	(1 word)	0H (0)
ACH (172)		Area-4	(1 word)	0H (0)
ADH (173)		Area-5	(1 word)	0H (0)
AEH (174)		Area-6	(1 word)	0H (0)
AFH (175)		Area-7	(1 word)	0H (0)
B0H (176)		Area-8	(1 word)	0H (0)
B1H (177)		Area-9	(1 word)	0H (0)
B2H (178)		Area-10	(1 word)	0H (0)
B3H (179)		Area-11	(1 word)	0H (0)
B4 to 16FH (180 to 367)	System area (Use prohibited)		(188 words)	———

(Address)	Buffer Memory			Default Value
170H (368)	Number of times IP packets received		(1 word)	0H (0)
171H (369)	Number of times received IP packet discarded because of check sum error		(1 word)	0H (0)
172H (370)	Total number of transmitted IP packets		(1 word)	0H (0)
173 to 17FH (371 to 383)	System area (Use prohibited)		(13 words)	———
180H (384)	Total number of received ICMP		(1 word)	0H (0)
181H (385)	Number of times received ICMP packet discarded because of check sum error		(1 word)	0H (0)
182H (386)	Total number of transmitted ICMP packets		(1 word)	0H (0)
183H (387)	Total number of received ICMP echo request packets		(1 word)	0H (0)
184H (388)	Total number of transmitted ICMP echo reply packets		(1 word)	0H (0)
185H (389)	Total number of transmitted ICMP echo request packets		(1 word)	0H (0)
186H (390)	Total number of received ICMP echo reply packets		(1 word)	0H (0)
187 to 18FH (391 to 399)	System area (Use prohibited)		(9 words)	———
190H (400)	Total number of received TCP packets		(1 word)	0H (0)
191H (401)	Number of times received TCP packet discarded because of check sum error		(1 word)	0H (0)
192H (402)	Total number of transmitted TCP packets		(1 word)	0H (0)
193 to 19FH (403 to 415)	System area (Use prohibited)		(13 words)	———
1A0H (416)	Total number of received UDP packets		(1 word)	0H (0)
1A1H (417)	Number of times received UDP packet discarded because of check sum error		(1 word)	0H (0)
1A2H (418)	Total number of transmitted UDP packets		(1 word)	0H (0)
1A3 to 1BFH (419 to 447)	System area (Use prohibited)		(29 words)	———
1C0 to 1C1H (448 to 449)	Subnet mask field		(2 words)	0H (0)
1C2 to 1C3H (450 to 451)	Default router IP address		(2 words)	0H (0)
1C4H (452)	Registered number of routers		(1 word)	0H (0)
1C5 to 1C6H (453 to 454)	Router 1 setting	Subnet address 1	(2 words)	0H (0)
1C7 to 1C8H (455 to 456)		Router IP address 1	(2 words)	0H (0)
1C9 to 1CAH (457 to 458)	Router 2 setting	Subnet address 2	(2 words)	0H (0)
1CB to 1CCH (459 to 460)		Router IP address 2	(2 words)	0H (0)
1CD to 1CEH (461 to 462)	Router 3 setting	Subnet address 3	(2 words)	0H (0)
1CF to 1D0H (463 to 464)		Router IP address 3	(2 words)	0H (0)
1D1 to 1D2H (465 to 466)	Router 4 setting	Subnet address 4	(2 words)	0H (0)
1D3 to 1D4H (467 to 468)		Router IP address 4	(2 words)	0H (0)
1D5 to 1D6H (469 to 470)	Router 5 setting	Subnet address 5	(2 words)	0H (0)
1D7 to 1D8H (471 to 472)		Router IP address 5	(2 words)	0H (0)
1D9 to 1EFH (473 to 495)	System area (Use Prohibited)		(23 words)	———
1F0H (496)	Communication specification during STOP		(1 word)	0H (0)
1FI to 1FFH (497 to 511)	System area (Use prohibited)		(15 words)	———

(Address)	Buffer Memory		Default Value
200H (512)	Transmission data length storage area/ reception data length storage area	Fixed buffer No.1/ Fixed buffer No.5 (1024 words)	0H (0)
201 to 5FEH (513 to 1535)	Transmission data write area/ reception data storage area		
600H (1536)	Transmission data length storage area/ reception data length storage area	Fixed buffer No.2/ Fixed buffer No.6 (1024 words)	
601 to 9FFH (1537 to 2559)	Transmission data write area/ reception data storage area		
A00H (2560)	Transmission data length storage area/ reception data length storage area	Fixed buffer No.3/ Fixed buffer No.7 (1024 words)	
A01 to DFFH (2561 to 3583)	Transmission data write area/ reception data storage area		
E00H (3584)	Transmission data length storage area/ reception data length storage area	Fixed buffer No.4/ Fixed buffer No.8 (1024 words)	
E01 to 11FFH (3585 to 4607)	Transmission data write area/ reception data storage area		
1200 to 1DFFH (4608 to 7679)	Buffer for random access	First half/last half (3072 words)	

Point

The following table shows the relationship of the parameters and functions that must be set when using the E71 functions.

Parameters \ Functions	Fixed Buffer Exchange: With Procedure	Fixed Buffer Exchange: Without Procedure	Random Access Buffer Exchange	Reading and Writing Data in the PLC CPU	Router Relay
Parameters for initial processing			———		○
Local station IP address	○	○	○	○	○
Special function settings	×	×	×	×	○
Various timer values	○	○	○	○	
Exchange parameters			———		
Usage applications			———		×
Bit 0 (Buffer application)	○	○	×	×	×
Bit 1 (Existence check)	△	△	△	△	×
Bit 7 (Pairing)	△	△	×	×	○
Bit 8 (Communication format)	○	○	○	○	×
Bit 9 (Exchange procedure)	○	○	×	×	○
Bit 14 · 15 (Open method)	○	○	○	○	
Exchange address *1					
E71 port No. / Remote node IP address / Remote node port No. / Remote ethernet address	○	○	○	○	×
Routing information	×	×	×	×	○

○ : Setting is required when using these functions (default value/change value)
× : Setting not required

*1 The parameters are set by the open processing communications format (TCP/UDP [Set with the exchange parameter usage application setting bits 14, 15]) when the communication line is connected.

Parameters \ Communication Format Open Method	TCP (00/01/11): Active (00): Remote Node ARP Functions: Yes	TCP: Active (00): Remote Node ARP Functions: No	TCP: Passive (01/11): Unpassive (01)	TCP: Passive (01/11): Full passive (11)	UDP (00): Remote Node ARP Functions: Yes	UDP (00): Remote Node ARP Functions: No
E71 port No.	○	○	○	○	○	○
Remote node IP address	○	○	×	○	○	○
Remote node No.	○	○	×	○	○	○
Remote node ethernet address	○**	○	×	×	○**	○

** Please make the default value (FFFFFFFFFFFFH)

MEMO

4. SETTINGS AND PROCEDURES UP TO OPERATION

This section explains the procedures and the setting methods for the system that uses E71 up to the point of E71 operation.

4.1 Abbreviated Procedures Up to Operation

Procedures up to Operation

1. Install the E71 into the base unit.
2. Lay out the connection cable and connect it to the remote node. .... Refer to Item 4.7.
3. Set the conversion switch for the interface between 10BASE5 (Ethernet) and 10BASE2 (Cheapernet) on the front of E71. .... Set only for AJ71E71-S3 (Setting not necessary for A1SJ71E71-B2/B5-S3) refer to Item 4.3.2.
4. Set using the exchange condition setting switch on the front of the E71. .... Refer to Item 4.3.2.
5. Perform a self diagnostic test.
   · RAM Test · ROM Test · Self Wrap Test .... Refer to Item 4.6.
6. Set the operation mode setting switch to 0 (on line) and reset the sequence for CPU. .... Refer to Item 4.3.1.
7. · Perform initial processing.
   · Perform open processing. .... Refer to Items 5.2 to 5.5.
8. Perform wrap test. .... Refer to Item 4.8 and Item 10.7.
9. Use the sequence program to set the E71 parameters and use a simple check program to check that the data is transmitted and received normally. ....
   - Fixed buffer exchange ——— Chapters 6 and 7
   - Random access buffer exchange ——— Chapter 8
   - Reading and writing data in PLC CPU — Chapters 9 and 10

DANGER

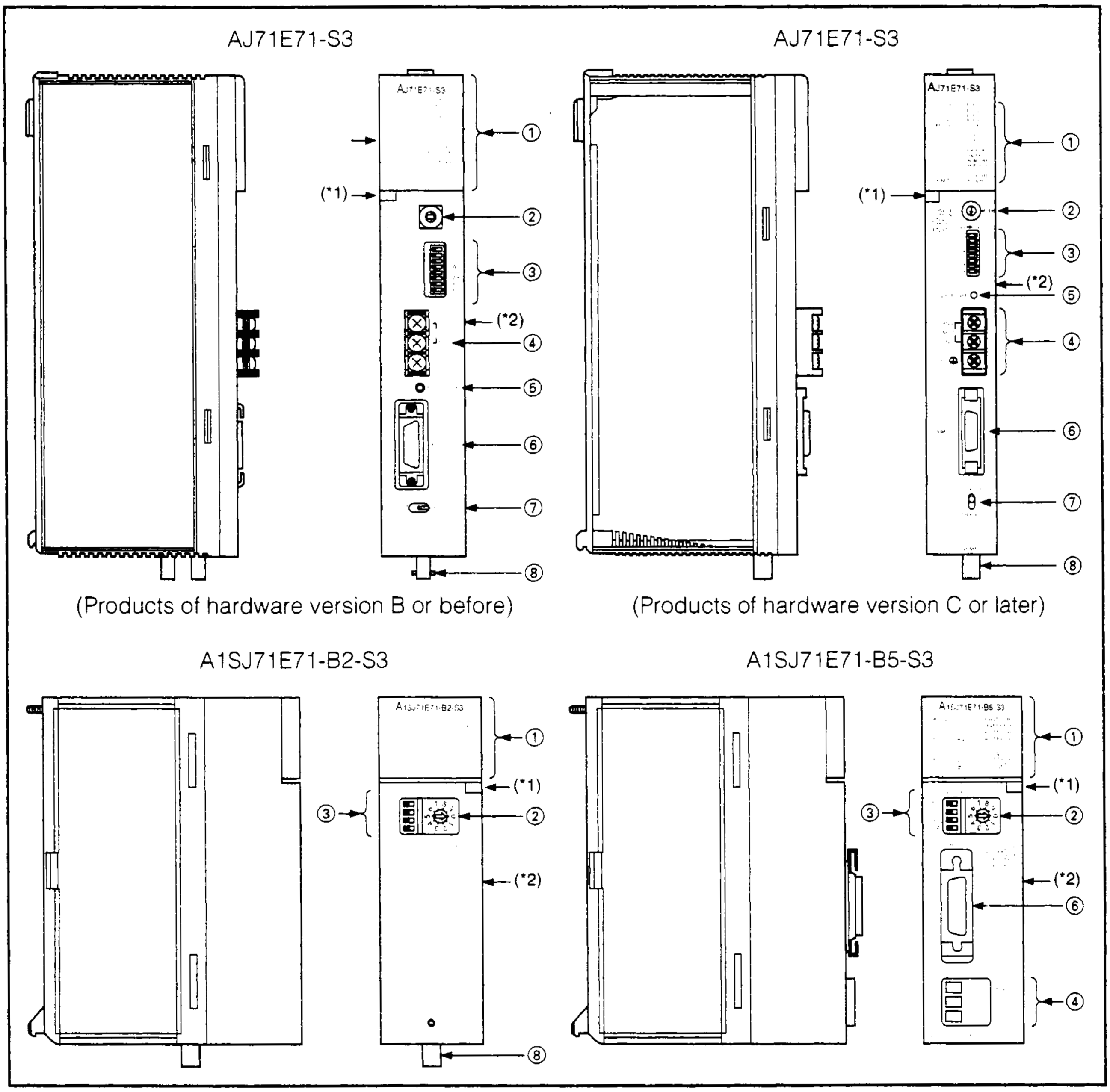
- Do not touch the terminals while the electricity is on.
  Doing so could cause erroneous operation.
- Make sure to switch all phases of the external power supply off before cleaning or re-tightening screws.
  If you do not switch off the external power supply, it will cause failure or erroneous operation of the module.
  If the screws are loose, it may result in fallout, short circuits, or erroneous operation.
  Tightening the screws too far may cause damage to the screws and/or the module, resulting in fallout, short circuits, or erroneous operation.

CAUTION

- Do not disassemble or modify the module.
  It may cause failure, erroneous operation, injury, or fire.
- Make sure to switch all phases of the external power supply off before cleaning or removing the module.
  If you do not switch off the external power supply, it will cause failure or erroneous operation of the module.

4.2 Names of Parts

This section explains the names and setting method for the E71 parts.



*1 It is a seal to show a hardware version and software version of a module.

(Example)

A B

- B — Software version shows "B"
- A — Hardware version shows "A"

*2 Date column of the rated plate shows the manufactured date of the module (Year (last two digits), month (two digits)) and the function version (one digit).

(Example)

98 09 B

- B — Function version (one digit) ····Printed only on products of version B or later
- 09 — Month (Two digits)
- 98 — Year (last two digits)

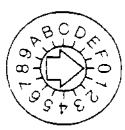
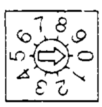
No.	Name	Validity by function AJ71 E71-S3	A1SJ71 E71-B2 -S3	A1SJ71 E71-B5 -S3	Description and Explanation	Reference Item
①	Display LED	○	○	○	Displays the operating state, data transmission and reception, and error description.The description when the LED is turned on or off varies for each LED.	Item 4.4
②	Operation mode setting switch	○	○	○	Select from on-line, off-line, self diagnostic test, or operation mode. Normally on-line is selected. The setting at the time of factory shipment is 0 (on-line).	Item 4.3.1
③	Exchange condition setting switch	○	○	○	Selects the start up conditions, exchange processing conditions, code types during exchange, and whether there is TCP time out error processing. The setting at the time of factory shipment is that SW1 to 8 /SW 1 to 4 are all turned off.	Item 4.3.2
④	External power supply terminal	○	×	○	Power supply terminal for supplying power to the transceiver when used as 10BASE5. Not required when used as 10BASE2.	Item 2.3
⑤	External power supply on indicator light	○	×	×	This light confirms that power is being supplied to the transceiver when used as 10BASE5. Not necessary when used as 10BASE2.	Item 2.3
⑥	AUI cable connection connector	○	×	○	Connector for connecting the E71 to the 10BASE5.	Item 2.3 Item 4.7.2
⑦	10BASE5 and 10BASE2 change over switch	○	×	×	Use this switch to change the interface between 10BASE5 and 10BASE2. The setting at the factory shipment time is 10BASE5.	—
⑧	10BASE2 connection connector	○	○	×	This connector is used to connect the QE71 to the 10BASE2.	Item 2.3 Item 4.7.3

○ : Yes × : No

4.3 Switch Settings

4.3.1 Operation Mode Settings

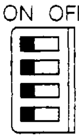
Table 4.1 List of Operation Mode Settings and Descriptions

Operation Mode Setting Switches	Setting No.	Setting Name	Settings Description
(AJ71E71-S3)  (A1SJ71E71-B2/B5-S3) 	0	On-line	Conducts exchange with remote node in the normal operation mode.
	1	Off-line	Disconnects the local station from the network.
	2	Test 1	Conducts a self diagnostic test using a self wrap test.
	3	Test 2	Conducts an RAM test.
	4	Test 3	Conducts a ROM test.
	5 to F/9	Usage Not Possible	

Point

When changing the operation mode, reset the PLC CPU after changing the operation mode setting switch. The resetting operation will start up the selected operation mode.

4.3.2 Exchange Condition Settings

Exchange Condition Setting Switch	Switch AJ71 E71-S3	Switch A1SJ71 E71-B2 /B5-S3	Setting Items	Setting Description	At Time of Factory Shipment
(AJ71E71-S3) OFF ON 1 2 3 4 5 6 7 8  or SW→ON 1 2 3 4 5 6 7 8  (A1SJ71E71-B2/ B5-S3) ON OFF SW1 SW2 SW3 SW4 	1	SW1	Line processing selection during TCP time out error	Selects the line processing when the TCP or ULP time out error occurs. Off: Closes the line when the TCP or ULP time out error occurs. On: Does not close the line even if a TCP or ULP time out error occurs.	OFF
	2	SW2	Data code setting	Selects the type of data code for exchanging data with the remote node. (Refer to Item 3.3) Off: Conducts exchange in binary code. On: Conducts exchange in ASCII code.	OFF
	3	—	Usage not possible		OFF
	4				OFF
	5				OFF
	6				OFF
	7	SW3	CPU exchange timing setting	Selects whether to approve or forbid data arriving from an external node when a PLC CPU is running. (While exchanging data read/write in the PLC CPU.) Off: Forbids writing from a remote node when the PLC CPU is running. On: Conducts writing from the remote node when the PLC CPU is running.	OFF
	8	SW4	Initial timing setting	Selects the initial processing start up timing. Off: Quick start (starts without a delay time) Set when one network is used for the entire configuration. On: Normal start (starts after a delay of 20 seconds) Use when the entire configuration is made up of multiple networks.	OFF

(1) Line processing selection using TCP time out error

A TCP or ULP error occurs when an ACK is not returned, even when the specified retry processing is conducted when using the TCP protocol. Selects the connection processing at this time.

When setting the SW1 to on and a TCP/ULP timeout error is detected by the E71 when data is exchanged, the remote node connection will be closed (line disconnected) and it cannot be reopened. Make a selection after checking the remote node specifications.

(2) Data code setting

Selects the data code type (binary, ASCII) when conducting data exchange with a remote node. (Refer to item 3.3)

(3) Initial timing setting

Freezes for approximately 20 seconds a connection that has been closed once when using TCP/IP protocol. Because there will be a wait when the same IP address and the same port No. are reopened, the system start up time should be set in anticipation of this. This setting will change the time from when the initial request signal (Y19) is turned on, to when the initial normal end signal (X19) is turned on.

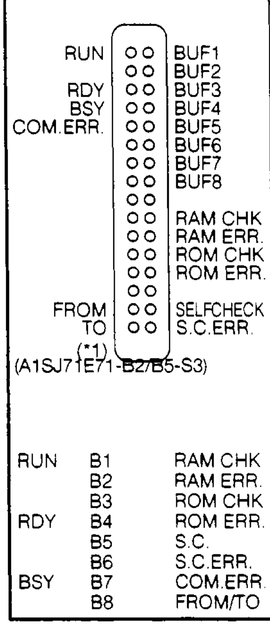
> **Point**
>
> The exchange condition setting switches should be set when the E71's power is turned off.

4.4 Description of Display LED's Display

This section explains the signal names and describes the displays of the display LEDs on the top front of the E71.

Table 4.2 List of the Display LEDs and the Display Contents

Display LEDs



LED Names: AJ71 E71-S3	LED Names: A1SJ71 E71-B2 /B5-S3	Description of LED Displays		When the LED is On	When the LED is Off
RUN		Normal operation display		Normal	Error
RDY		Exchange preparation completed display		Flashes after initial processing finishes normally	
BSY		Exchange processing executing display		Executing	Not executing
COM. ERR.		Exchange error detection display (Refer to Item 13.2*1)		During error detection	Normal (no error)
FROM	FROM/TO	Display during data read (FROM) / display during data write (TO)		During read/write processing execution	Read/write is not being processed
TO or FROM/TO	—				
BUF 1	B1	Connection No.1	Communication line connection state	Open completed	Closed state
BUF 2	B2	Connection No.2			
BUF 3	B3	Connection No.3			
BUF 4	B4	Connection No.4			
BUF 5	B5	Connection No.5			
BUF 6	B6	Connection No.6			
BUF 7	B7	Connection No.7			
BUF 8	B8	Connection No.8			
RAM CHK	RAM CHK	RAM test executing display		Test executing	Not being tested
RAM ERR.	RAM ERR.	RAM error detection display		RAM error	Normal
ROM CHK	ROM CHK	ROM test executing display		Test executing	Not being tested
ROM ERR.	ROM ERR.	ROM error detection display		ROM error	Normal
SELFCHECK	S.C.	Self loopback test executing display		Test executing	Not being tested
S.C. ERR.	S.C. ERR.	Self loopback error detection diaplay		Loopback error	Normal

*1 The TO LED in the above figure is replaced with the FROM/TO LED in AJ71E71-S3 hardware version C or later. The FROM LED is removed.

(1) For the above LEDs, when RUN turns off after the power is turned on, a watch dog timer error can be suspected. The watch dog timer error detection signal (X1F) will also turn on.

(2) The RDY in the above LED flashes when operation is started by the online mode (mode setting switch is set to 0) and the initial processing finishes normally.

(3) Of the above LEDs, the BSY exchange processing executing is done during the following times. BSY is lit while retrying processing during data transmission.

(a) For Transmission Procedure

- Until a reception response is received from the command transmission
- Until a transmission error occurs from the command transmission (until time out)

(b) For Reception Procedure

- Until a response is returned from the command reception

(4) For the LEDs shown in Table 4.2, the BUF1 to BUF8 /B1 to B8 communication line connection state shows the line connection state with the partner set by the exchange parameters. The open completed signal (X10 to X17) ON/OFF state can be checked using these LEDs. Here data can only be exchanged using connections that are opened.

(5) The COM.ERR LED can be turned off by output signal (Y17).

Point

In the explanation hereafter in this manual the display LED's LED names will be shown using the AJ71E71-S3 names.

4.5 Mounting and Installation

This section explains the handling precautions that are common for modules, and about the installation environment from the time the E71 is unpacked until installation.

For details regarding module mounting and installation, please refer to the user's manual of the PLC CPU module being used.

4.5.1 Handling Precautions

This section explains the handling precautions for the E71 module itself.

For the cautionary items regarding module installation and removal, refer to the ●Safety Precautions● described in the beginning of this manual.

(1) The E71's case is made of plastic, so do not drop it or subject it to strong impacts.

(2) The tightening torque for module terminal screws and installation screws should be kept within the following range.

(a) AJ71E71-S3

Screw Locations	Tightening Torque Range
Power supply line connection terminal screws (M4 screws)	98 to 137 N · cm
Module installation screws (normally not required) (M4 screws)	78 to 118 N · cm

(b) A1SJ71E71-B2-S3, A1SJ71E71-B5-S3

Screw Locations	Tightening Torque Range
Power supply line connection terminal screws	40 N · cm
Module installation screws (M4 screws)	78 to 118 N · cm

⚠CAUTION

- Insert the tabs at the bottom of the module into the mounting holes in the base unit before installing the module. (Modules in AnS series, make sure screws are securely tightened to base unit with specified torques.)
  Improper installation may cause erroneous operation, failure, or the module to fall out.
- Be sure that cuttings, wire chips, or other foreign matter do not enter the module. Foreign matter may start a fire or cause an accident or erroneous operation.
- Do not disassemble or rebuild the module. It may cause failure, erroneous operation, injury, or fire.
- Make sure to switch all phases of the external power supply off before mounting or removing the module.
  If you do not switch off the external power supply, it will cause failure or erroneous operation of the module.
- Tighten the terminal screws within the range of specified torque. If the screws are loose, it may result in short circuits or malfunctions.
  Tightening the screws too far may cause damage to the screw and/or the module, resulting in fallout, short circuits, or erroneous operation.
- Do not touch the electronic parts or the module conducting area.
  It may cause erroneous operation or failure.
- When disposing of this product, handle it as industrial waste.

4.5.2 Installation Environment

The following environments should be avoided when installing the QnA series PLC.

- Areas where the temperature range of the surrounding temperature is outside 0 to 55 degrees Celsius.
- Areas where the surrounding humidity exceeds the range of 10 to 90% RH
- Areas where there are sharp changes in humidity or where condensation forms
- Areas where there are corrosive gases or flammable gases
- Areas where there are conducting powders such as dust or iron dust, and where there is oil mist, salt, or organic solvents
- Locations that are struck by direct sunlight
- Areas where there are strong electric fields or strong magnetic fields
- Areas where direct vibrations or shocks will be transmitted to the module

⚠CAUTION	• Use the PLC in the environment given in the general specifications section of this manual. Using the PLC outside the range of the general specifications may result in electric shock, fires, or erroneous operation or may damage or degrade the product.

4.6 Self-Diagnostic Test

This section explains the self diagnostic test that is used to check the E71's transmission and reception functions and its hardware.

4.6.1 Self-Loopback Test

This section explains the self loopback test that is used to check the hardware that includes the E71's transmission and reception lines.

The self loopback test is a check that transmits a test message to the E71's own node via a line, and that can receive the same message via the network.

Following is an explanation of the self loopback test. This test is conducted in approximately 5 seconds.

Self wrap test method

① Connect the E71 to the line. (Refer to Item 4.7)

② Set the 10BASE5 and 10BASE2 changeover switch. (Only for AJ71E71-S3)

③ Set the operation mode setting switch on the front of the E71 to the 2 position.

④ Set the PLC CPU's RUN/STOP key switch to STOP.

⑤ When the PLC CPU is reset, the self loopback test will begin.
(The SELFCHECK LED is lit.)

Test results

The test results can be determined from the LEDs on the front of the E71.

① If the SELFCHK LED is turned off, then the self loopback test is completed.

② The test results can be checked using the S.C.ERR.'s LED.

When normal .... LED is turned off

When there is an error .... The LED is lit

③ The following can be suspected causes of errors.

- E71 hardware error
- Ethernet line error
- External supply power 12VDC's error (Only 10BASE5)

Operation after test is completed

The PLC CPU can be reset after the on-line mode or another test mode is changed over using the operation mode setting switch on the front of the E71.

Point

There will be no hardware interference if the self loopback test is conducted while another mode is on-line. If there is a packet in the line, this test may not finish within approximately 5 seconds if interference occurs with the packet. In this case, perform the test after terminating the data exchange with the remote node.

4.6.2 RAM Test

This section explains the RAM test that is performed to check the E71's RAM.

RAM test method

① Set the operation mode setting switch on the front of the E71 to the 3 position.

② Set the PLC CPU's RUN/STOP key switch to the STOP position.

③ When the PLC CPU is reset, the RAM test will begin.
(RAM CHK LED is lit.)

Test results

The test results can be determined from the LEDs on the front of the E71.

① If the RAM CHK's LED is not lit, the RAM test is completed.

② The test results can be checked using the RAM ERR.'s LED.

When normal .... The LED is not lit

When error .... The LED is lit

③ The following can be suspected as causes of an error.

- E71 hardware error
- RAM error

Operation after the test is completed

Reset the PLC CPU after changing the operation mode setting switch on the front of the E71 to either the on-line mode or the other test mode.

Point

When there is an error for the test results of the RAM test shown in this item, reconduct the same test.
If an error occurs a second time, then a E71 hardware error can be suspected. For details regarding troubles, please consult with you nearest branch or agent.

4.6.3 ROM Test

This section explains the ROM test that is used to check the E71's ROM.

ROM test method

① Set the operation mode setting switch on the front of the E71 to the 4 position.

② Set the PLC CPU's RUN/STOP key switch to the STOP position.

③ The ROM test will begin after the PLC CPU is reset.
(ROM CHK LED is lit.)

Test results

The test results can be determined from the LEDs on the front of the E71.

① If the ROM CHK LED is turned off, the ROM test is completed.

② The test results can be checked using the ROM ERR.'s LED.
When normal .... LED is turned off
When error .... LED is turned on

③ The following can be suspected as causes of an error.
- E71 hardware error
- ROM error

Operation after test is completed

Reset the PLC CPU after changing the operation mode setting switch on the front of the E71 to either the on-line mode or another test mode.

Point

If there is an error for the test results of the ROM test shown in this item, reconduct the same test. If an error is again generated, then a E71 hardware error can be suspected. For details regarding troubles, please consult with the branch office or agent nearest you.

4.7 Connecting to the Network

This section explains the method for connecting the E71 to the 10BASE5 or 10BASE2.

4.7.1 Connection Precautions

(1) Sufficient safety precautions are required when installing 10BASE5 and 10BASE2. Consult with a specialist when installing connection cable terminal processing or trunk line cables, etc.

(2) Use a connection cable that meets the standard shown in Item 2.3.

(3) The allowable bending radius for coaxial cables is set. When bending coaxial cables to connect them, a space that is larger than the coaxial cables' allowable radius is required. For information regarding the coaxial cables' allowable bending radius, please consult the cable manufacturer.

⚠CAUTION	• When installing AUI cables (transceiver cable)/coaxial cables, do not bundle them or place them close to main lines or power lines. Keep them at least 100mm(3.94 inch) away from such cables. Noise may cause erroneous operation. • Do not connect the AUI cable when the module installation station's power is turned on. • Be sure to fix communication cables and power cables leading from the module by placing them in the duct or clamping them. Cables not placed in the duct or without clamping may hang or shift, allowing them to be accidentally pulled, which may result in a module malfunction and cable damage. • When detaching the communication cable or power cable from the module, do not pull the cable portion. For cables with connectors, hold the connector at the junction to the module, then detach it. For connectors without connectors, first loosen the screw at the junction, then detach the cable. Pulling the cable portion while it is connected to the module may cause a malfunction or damage to the module and cable.

4.7.2 Connecting to 10BASE5

This section explains the method for connecting E71 to a 10BASE5 network.

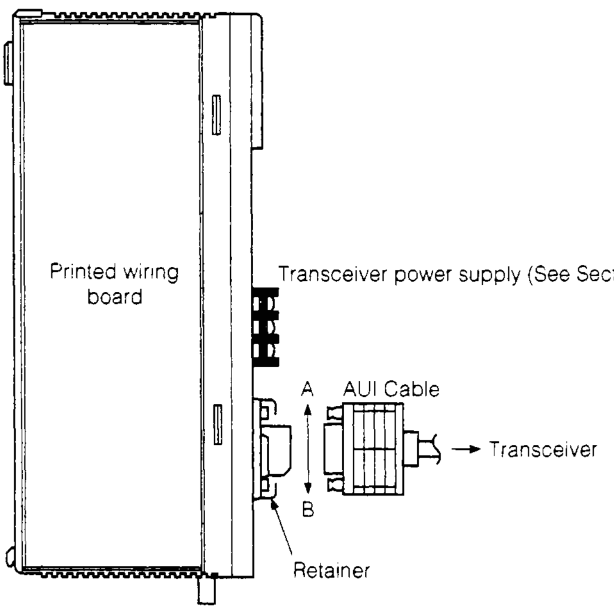


Fig 4.1 AUI Cable Connection Diagram

Method for connecting to the AUI cable (Transceiver cable)

Procedure	Description of operation	AJ71E71-S3 Products of H/W ver. C or later	AJ71E71-S3 Products of H/W ver. B or before	A1SJ71E71-B2/B5-S3
1	Slide the retainer in the A/B direction as shown in Figure 4.1.	A direction	B direction	B direction
2	Push the AUI cable connector to the back.	—	—	—
3	Slide the retainer in the A/B direction as show in Figure 4.1.	B direction	A direction	A direction
4	Check that the AUI cable is locked.	—	—	—
5	Turn on the transceiver's power.	—	—	—

⚠CAUTION • Do not connect the AUI cable when the module installation station's power is turned on.

Point

When connection to the network is made using the 10BASE5, if countermeasures against high-frequency and noise generated in the installation environment of E71 is necessary, attach a ferrite core to the transceiver side of the AUI cable to eliminate these effects.
Refer to the POINT column in Section 2.3.

4.7.3 Connecting to 10BASE2

This section explains the method for connecting the E71 to the 10BASE2 network.

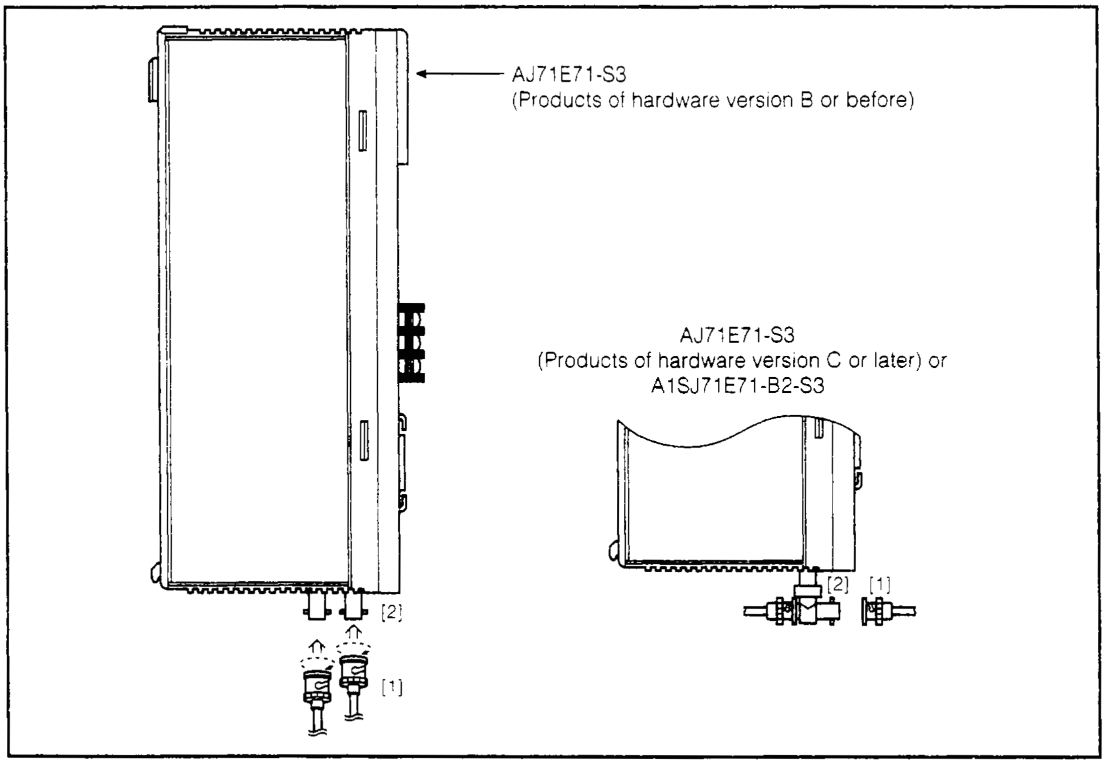


Fig 4.2 10BASE2 Coaxial Cable Connection Diagram

10BASE2 Coaxial cable connection method

(1) As shown in Figure 4.2, line up and push in the lip [2] into the groove [1].

(2) Turn the connector 1/4 rotation to the right while pushing it in.

(3) Turn the connector until it locks.

(4) Check that the connector is locked.

Remarks

Coaxial cable connector connection

This section explains the method for connecting the BNC connector (coaxial cable connector plug) to the cable.

(1) BNC connector and coaxial cable configuration

Fig 4.3 shows the BNC connector and coaxial cable configuration.

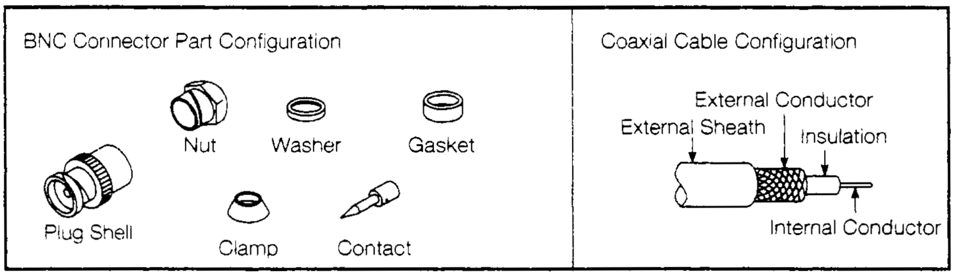
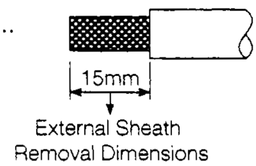


Figure 4.3 BNC Connector and Coaxial cable configuration

(2) Method for connecting the BNC connector and the coaxial cable

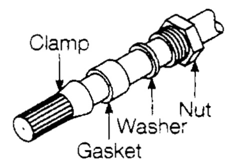
The following shows a method for connecting the BNC connector to the coaxial cable.

(a) Remove the coaxial cable's external sheath as shown in the drawing at right. Be careful not to damage the external conductor.



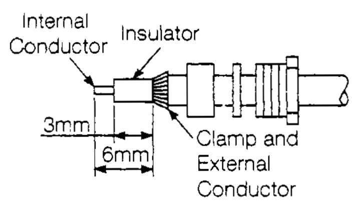
↓

(b) Place the nut, washer, gasket, and clamp on the coaxial cable as shown in the drawing at right, and then wrap the external conductor.



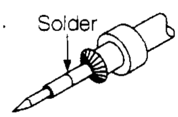
↓

(c) Cut the external conductor, insulator, and internal conductor to the dimension as shown in the drawing at right. However, to cut the external conductor to the same dimension as the clamp's tapered portion, place the clamp on before cutting.



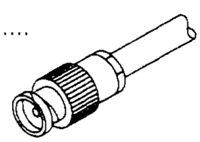
↓

(d) Solder the connector to the internal conductor.



↓

(e) Insert the conductor assembly in (d) into the plug shield, and screw on the plug shield nut.



Point

Take the following precautions when soldering the contact to the internal conductor.

(1) Be sure that the solder does not creep up the soldered area.
(2) Be sure that there are no gaps in or biting into the conductor and cable's insulation.
(3) Quickly do the soldering to prevent the insulation from deforming.

4.8 Loopback Test

The loopback test is a function that tests whether exchange is conducted normally between a node and the local station E71. When a loopback test is conducted, the data transmitted from the remote node is then retransmitted as is by the E71 as a response to the originating station. A loopback test will be conducted using the function shown in Item 10.7 after the initial processing and open processing finish normally.

4.9 Maintenance and Inspection

There are no inspection items for the E71 other than the terminator and checking the cable connections for looseness. In addition to this, to keep the system in good condition at all times and conduct the inspection items contained in the PLC CPU's module's user manual.

DANGER

- Do not touch the terminals while the electricity is on. Doing so could cause erroneous operation.
- Make sure to switch all phases of the external power supply off before cleaning or re-tightening screws.
  If you do not switch off the external power supply, it will cause failure or erroneous operation of the module.
  If the screws are loose, it may result in fallout, short circuits, or erroneous operation. Tightening the screws too far may cause damage to the screw and/or the module, resulting in fallout, short circuits, or erroneous operation.

CAUTION

- Do not disassemble or rebuild the module. It may cause failure, erroneous operation, injury, or fire.
- Make sure to switch all phases of the external power supply off before mounting or removing the module.
  If you do not switch off the external power supply, it will cause failure or erroneous operation of the module.
- Do not touch the electronic parts or the unit conducting area.
  It may cause erroneous operation or failure.

MEMO

5. PROCEDURES FOR EXCHANGING WITH REMOTE NODES

5.1 Overview of Exchange Procedures

This section shows the general procedure used to exchange data between the E71 and a remote node.

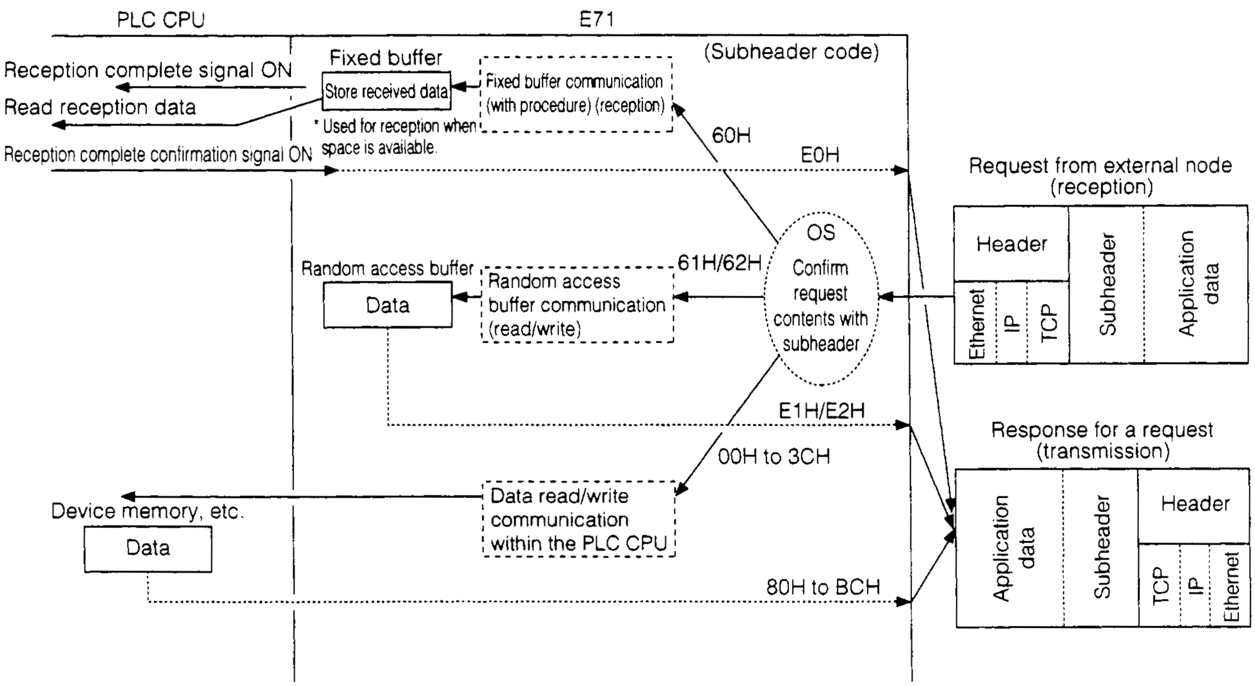
To begin data exchange, the initial processing and open processing must be used to connect with the exchange partner and the communication line.

To end data exchange, conduct close processing and end processing. This disconnects the communication line and ends all exchange processing.

Point

(1) When conducting either fixed buffer exchange, random access buffer exchange, or reading and writing to the PLC CPU, open processing must be conducted with an exchange remote node. In addition, all of the above three types of exchange can be conducted with user opened remote nodes.

When receiving communication request data from an external node



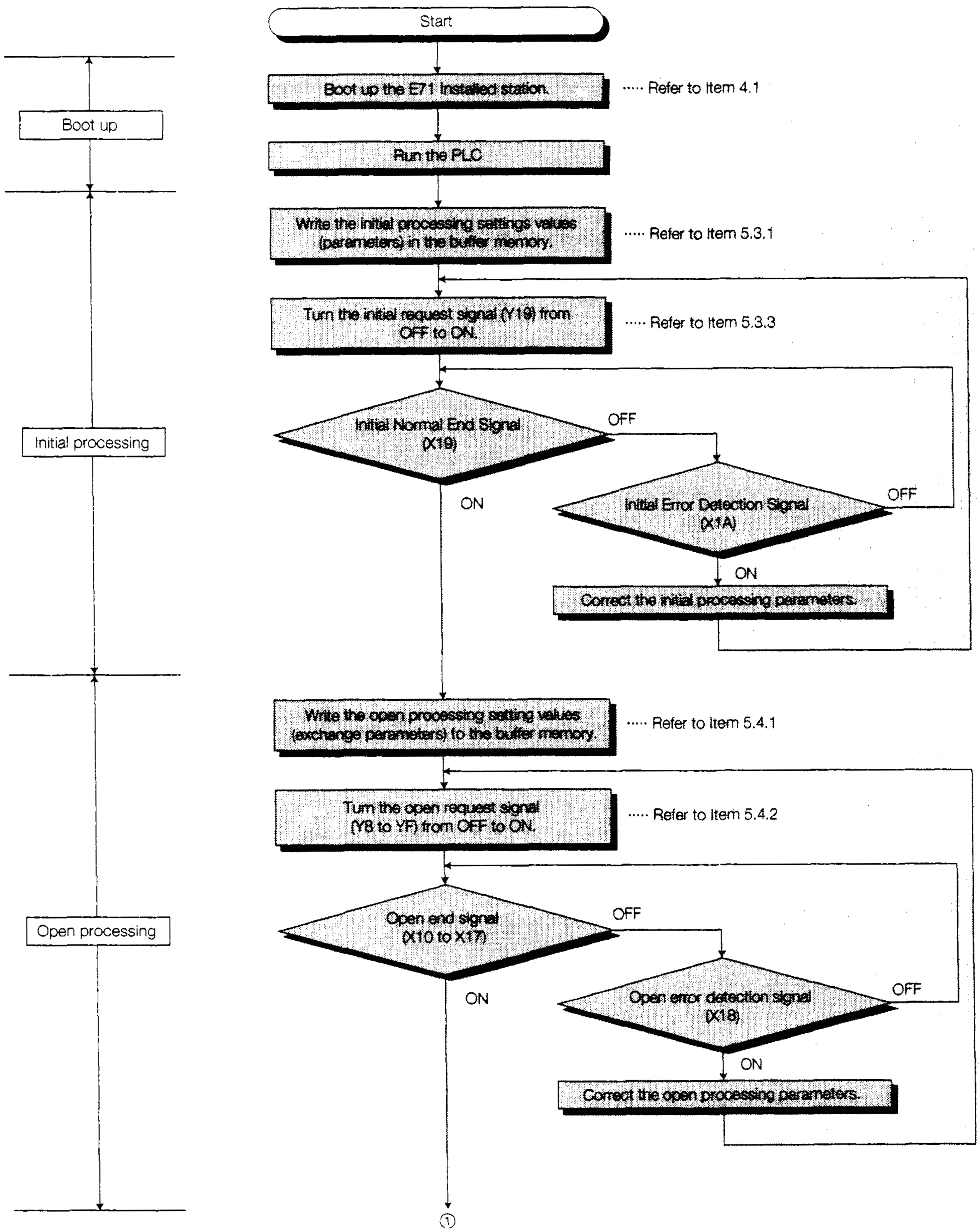
(2) Open processing can be conducted for a maximum of eight nodes. However, when translating and receiving with the same remote node using a fixed buffer, two fixed buffers are required so the number of nodes to which exchange can be conducted is reduced.

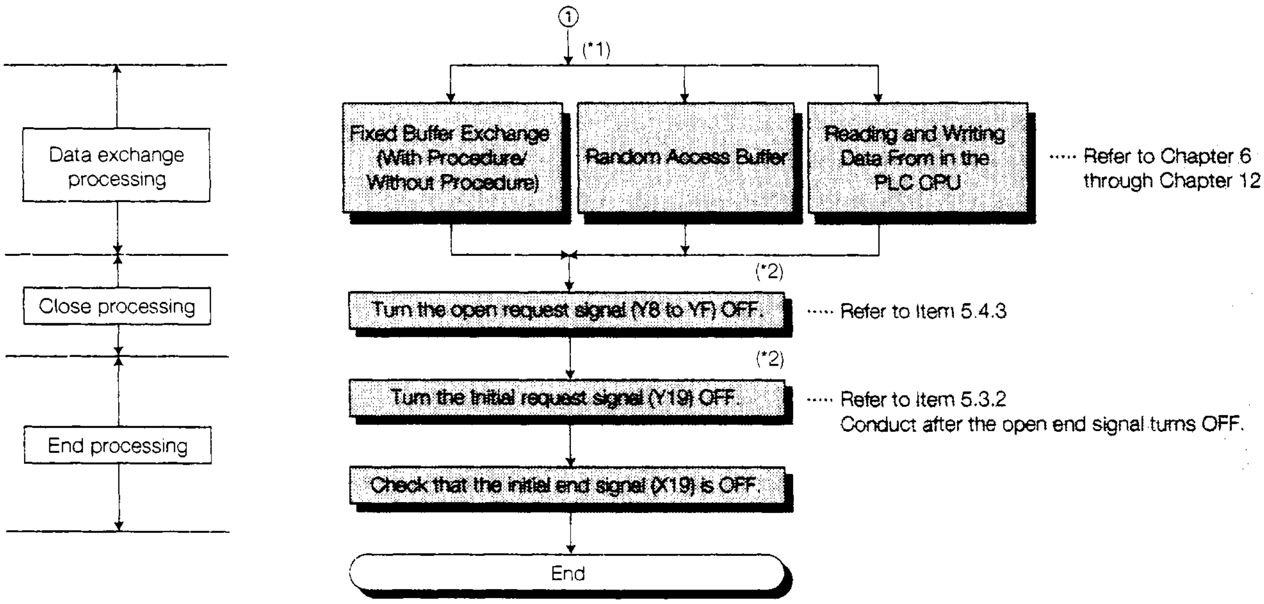
(3) The following explains the initial processing and the open processing when exchange prohibited is set by the Exchange Specification During STOP using the E71 buffer memory (address: 496).

- When an E71 installation station's PLC CPU is in the STOP state, the E71's open request signal (Y8 to YF) and initial request signal (Y19) are also turned off and the line to the remote node is closed.
- When the E71 installed station's PLC CPU is changed from STOP to RUN, reconduct initialization processing and open processing.

(Exchange procedures)

Initial processing, open processing, and data exchange PLC programs for the exchange procedure after the E71's installed station's PLC CPU write processing is completed.





*1 Once a communication line is connected, the following data communication can be performed between E71 and other destination nodes using the port number specified by the communication parameter during the open processing.

① When "procedural fixed buffer communication" is specified during the open processing (When bit 9 of the usage setting communication parameter is OFF (procedural))

Functions that can be communicated using the applicable connection [Refer to Item 3.5.2 for data transmission procedures.]	1	Communication using fixed buffers (Refer to Chapter 6.) (a) Either transmission or reception is possible. * This is determined based on the setting for bit 0 of the usage setting communication parameter. • When bit 0 of the usage setting is OFF, transmission is enabled. • When bit 0 of the usage setting is ON, reception is enabled. (b) Data transmission or reception between the PLC CPU and other nodes is performed using a fixed buffer (buffer memory) of the same number as the connection number of the open request signal sent when the communication line was opened. * The connection number of the open request signal is compatible with the fixed buffer number. (c) When transmitting and receiving data between the E71 and other nodes, two communications lines are required.
	2	Communication using random access buffer (Refer to Chapter 8.) Data is read and written from/to the random access buffer of E71.
	3	Read/write communication with respect to the data within the PLC CPU. (Refer to Chapters 9 and 10.) Data is read and written from/to the device memory of PLC CPU.

② When "non-procedural fixed buffer communication" is specified during the open processing (When bit 9 of usage setting communication parameter is on (non-procedural))

- Communication (transmission or reception) can be performed only by using the fixed buffer (Refer to Chapter 7).
- The number of fixed buffers used and the number of communication lines required for data transmission and reception are the same as those required for procedural fixed buffer communication.

*2 When exchange enable is set using Exchange Specification During STOP using the E71 buffer memory (address: 496) when the open request signal (Y8 to YF) and the initial request signal (Y19) are off, the following data exchange can be continued.

- Exchange using the random access buffer
- Exchange of data read/write in the PLC CPU

Continue data exchange in accordance with Item 5.6.

5.2 Connecting and Disconnecting Communication Lines

The arrangement between the nodes makes it necessary when beginning data exchange to connect a communication line between exchange partners, and when the data exchange is completed, to disconnect the communication line between exchange partners.

Following is an explanation of the connecting and disconnecting of E71 communication lines and of the initial processing setting data for exchanging data between the E71 and a remote node.

1 Connecting communication lines (Initial processing, open processing)

(a) Conduct E71 initial processing and open processing with the remote node and connect to the communication line using the parameters and switch settings specified by the user. Exchange can only be conducted with the remote mode for which the line was connected using this initial processing and open processing.

(b) Exchanging with a remote node using the fixed buffer, exchange using the random access buffer, and reading/writing the data in the PLC CPU is possible using the port No. specified during open processing when the communication line is connected.

2 Communication line disconnect (Close processing, end processing)

When data exchange with a remote node has been completed after a communication line was connected, the communication line is disconnected .

(a) Disconnection of the communication line by user processing
close processing and end processing are conducted by the PLC CPU.

(b) Disconnection of the communication line by error occuring
If the condition described in Item 3.5.3 occurs, the line will be forcefully closed.
End processing is conducted by the PLC CPU.

Point

(1) Number of nodes that can be exchanged
① When using E71 of which software version is "K" or before
- The maximum number of remote nodes for which exchange is possible during one initial processig for the E71 is 20 stations.
(The original station is included when a message is sent to several stations at once by UDP/IP.)
- If this is exceeded, an error (error code : A00EH, A00FH) will occur during open processing.
When this error occurs, initial processing will be reconducted after end processing and close processing of the data exchange for all connection currently opened are conducted and the initial request signal (Y19) turns off.

② When using E71 of which software version is "L" or later
There are no restrictions in number of remote nodes for which exchange is possible during one initial processing for the E71.

(2) Data exchange during the PLC CPU is in the STOP status
① The following data exchange can be continued even when the PLC CPU of the station installed in the E71 is in the STOP status by setting exchange enable at the "Exchange Specification During STOP" using the E71 buffer memory (address: 496).
- Exchange using random access buffer
- Read/write exchange of the data in the PLC CPU

② Continue data exchange or conduct reopening in accordance with Item 5.6.

5.3 Initial Processing and End Processing

This section explains about the E71 initial processing and end processing conducted by the PLC CPU when connecting to a communication line and exchang data with a remote node.

Connect to the line using the normal online operation.

5.3.1 Data for Initial Processing

This section explains the parameter setting area for conducting initial processing. The network manager (the person who plans the network and manages the IP addresses) writes the values to be used in this area before initial processing is conducted. (Refer to Point at the end of Item 3.7.2 about whether the parameter should be set.)

Buffer Memory

(Address)	Initial Processing Parameter Setting Area (16 Words)	Default Values
0 to 1H (0 to 1)	Local station E71's IP address (2 words)	0H (0)
2H (2)	Special function settings (1 word)	0H (0)
3H (3)	Timer setting time units (1 word)	7D0H(2000)
4 to 6H (4 to 6)	System area (Use prohibited) (3 words)	—
7H (7)	Destination existence check start interval timer value (1 word)	12CH (300)*1
8H (8)	Destination existence check interval timer value (1 word)	5H (5)*1
9H (9)	Number of retransmit tries for destination existence check (1 word)	3H (3)
AH (10)	TCP/ULP time out value (1 word)	FH (15)*1
BH (11)	TCP zero window timer value (1 word)	5H (5)*1
CH (12)	TCP retransmit timer value (1 word)	5H (5)*1
DH (13)	TCP end timer value (1 word)	AH (10)*1
EH (14)	IP setup timer value (1 word)	3H (3)*1
FH (15)	Response monitoring timer value (1 word)	FH (15)*1

*1 The setting of value units during default 2s can be changed to 500 ms.
(Timer value = setting value × 2 s or 500 ms)

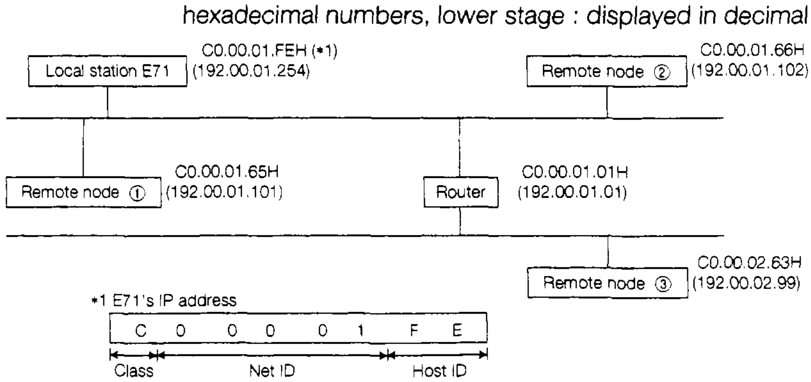
1 Local station E71's IP address (Default Value = 0H) .... Address 0H to 1H (0 to 1)

(a) The local station E71's IP address is set following the standard IP address (Refer to Item 11.3).

① Set it so that the local station E71 and the partner remote node to which exchange is being conducted are set to the same class net ID. In the following example, the E71 and the partner remote node ① ② IP address class net ID is set at "C00001□□H." (The host ID can be freely set to any No. other than "00H" and "FFH.")

② When the IP address net ID's of the local station E71 and the partner remote node to which exchange is being conducted are not the same, a setting for using the router relay function is required (Refer to Chapter 12). In the following example, the setting values (parameters) used for the router relay function when exchange is conducted between the E71 and the remote node ③ are set in the local station E71.

(Example) When the local station E71's IP address is class C (upper level: displayed in hexadecimal numbers, lower stage : displayed in decimal numbers).



(b) Except when a router relay function (Refer to Chapter 12) , the IP addresses can be freely allocated as described in (a) above.

Remarks

When the router relay function is used, please use addresses that conform with the standard IP addresses used on the global scale.
* Standard IP Address
IP addresses are divided into classes to allow an address system that corresponds to the size of the network to be selected. (Refer to Item 11.3)

2 Special function setting (Default value = 0) .... Address 2H (2)

(a) Sets whether a router relay function is used.
(Static router relay)
(b) Validates the buffer memory's subnet mask setting area and routing information area (address 448 to 472) setting value when set to use the router relay function.

(Bit Position)	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	0															①

① Router relay function setting (b0)
0 : Not used (Default value)
1 : Used

3 Timer setting time units (Default value = 7D0H (2000)) .... Address 3H (3)

(a) The timer value units set to the buffer memory address 7 to 15 can be set to either 500ms unit or 2s unit. (Default value 7D0H (2000) is shown as the 2s unit.)
(b) Specify the setting value as 1F4H (500) or other than 1F4H.
1F4H (500) : 500 ms unit
Other than 1F4H : 2s (2000ms) unit
(c) The timer values set by the buffer memory addresses 7 to 15 using the timer setting time units are specified for the following ranges.

Timer setting time units	Timer setting values setting possible range	Timer time range
2s (2000ms)	1 to 8191 (1 to 1FFFH)	2.0s to 16382.0s
500ms	1 to 32767 (1 to 7FFFH)	0.5s to 16383.5s

* Current timer operation cannot be guaranteed if setting values outside the above range are set.

(d) The timer times set in buffer memory addresses 7 to 15 are as follows.
Timer time=Timer setting value × timer setting time units
(Example) When the TCP/ULP time out value setting value is 15
① When the timer setting time unit is 2s : 15 × 2s = 30s (30000ms)
② When the timer setting time unit is 500ms : 15 × 500ms = 7500ms

4 Destination existence check begin interval timer value (Default value = 12CH (300), setting time = setting value × (Unit)) .... Address 11H (17)

(a) When the exchange with the partner remote node by the connection opened by destination existence check is finished until existence check is begun.
(b) The setting value is set to 1H to 1FFFH/1H to 7FFFH by the timer setting time's unit setting. (10 minutes when the default value is 12CH(300)) (*2)

5 Destination existence check interval timer value (Default value = 5H (5), setting time = setting value × (Unit)) .... Address 8H (8)

(a) Sets the interval time for conducting retry and existence check when no response is received from the partner remote node that is conducting the existence check for the connection opened by the destination existence check.
(b) The setting value is set to 1H to 1FFFH/1H to 7FFFH by the timer setting time's unit setting.

6 **Number retries for destination existence check (Default value = 3H (3)) .... Address 9H (9)**

(a) Sets the number of retries and existence checks when a response is not received from the partner remote node that is conducting the existence check for the connection that was opened by the destination existence check.

(b) The setting value is set to 1H to 7FFFH by the timer setting time's unit setting.

7 **TCP/ULP time out value (Default value = FH (15), setting time = setting value × (Unit)) .... Address AH (10)**

(a) Sets the pocket existence time during TCP data transmission.
This timer receives parameter when sending data and TCP open, and works for ARP function's existence time.

(b) The setting value is set to 1H to 1FFFH/1H to 7FFFH by the timer setting time's unit setting. (*1) (*3)

8 **TCP zero window timer value (Default value = 5H(5), setting time = setting value × (Unit))Address BH (11)**

(a) The window shows the reception buffer on the reception receiving end.

(b) When there is no more space in the reception buffer on the receiving end (window size = 0), the transmitting end waits to transmit data until there is space in the reception buffer on the receiving end. At this time, the receiving end follows the TCP zero window timer value to transmission window check packet to the reception end to check the receiving possibility condition.

(c) The setting value is set to 1H to 1FFFH/1H to 7FFFH by the timer setting time's unit setting. (*3)

9 **TCP retransmit timer value (Default value = 5H (5), setting time = setting value × (Unit)) .... Address DH (13)**

(a) If ACK is not returned during TCP open and data transmission, the retransmission time will be set. This timer sets retransmission time for ARP request if response to transmitted ARP request is not returned. (ARP's retransmission is conducted with TCP retransmission timer value /2.)

(b) The setting value is set to 1H to 1FFFH/1H to 7FFFH by the timer setting time's unit setting. (*1) (*3)

10 **TCP end timer value (Default value = 3H(3), setting time = setting value× (Unit)) .... Address DH (13)**

(a) Sets the monitoring time when waiting for FIN to be received from the partner node after the local station has transmitted FIN and ACK has been received from the partner node when the local station closes the TCP connection.

(b) When FIN is not received from the partner node after the TCP end timer time, RST is transmitted to the partner node to forcefully close the line.

(c) The setting value is set to 1H to 1FFFH/1H to 7FFFH by the timer setting time's unit setting. (*3)

11 **IP set up timer value (Default value = 3H(3), setting time = setting value × (Unit)) .... Address EH (14)**

(a) Exchange data is sometimes divided and transmitted by IP level due to the transmitting station's or the receiving station's buffer limitations. This sets the time to wait until the following divided data is restored (reassembled) when the E71 receives the divided data.

(b) The setting value is set to 1H to 1FFFH/1H to 7FFFH by the timer setting time's unit setting. (*3)

12 **Response monitor timer value (Default value = FH (15), setting time = setting value × (Unit)) .... Address FH (15)**

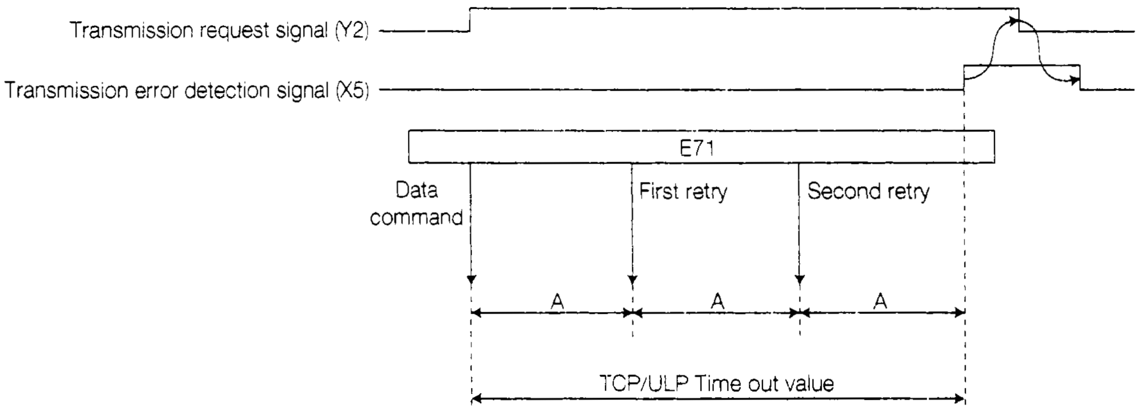
(a) Sets to the following time.
① The wait time from when a command is transmitted until a response is received.
② When a divided message is transmitted, the time from the first message transmission until the final message is received.

(b) The setting value is set to 1H to 1FFFH/1H to 7FFFH by the timer setting time's unit setting. (*3)

*1 When exchange errors occur due to noise, change the setting value to a higher number of retries. The retry number is determined using the following formula. (For the default value 2 = 15 ÷ 5 - 1)

Number of retries = (TCP/ULP time out value) ÷ (TCP retry timer value) - 1

(Example) When data cannot be transmitted when the setting value makes the number of retries 2, the transmission error detection signal will turn on with the timing shown in the diagram below. (When fixed buffer No.3 is used)



A : TCP retransmission timer value
(The data retransmission time when ACK is not returned after the data is transmitted)

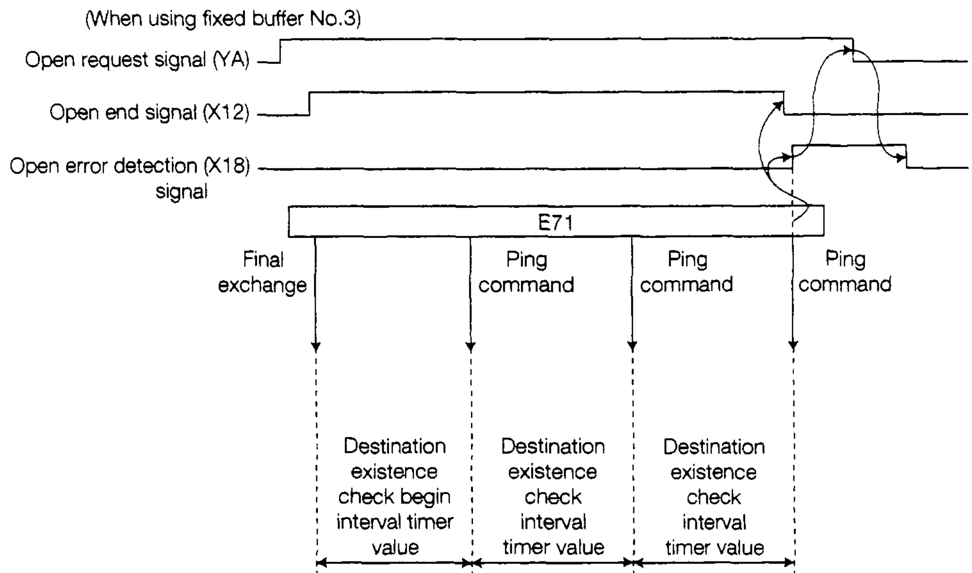
Remarks

When removing the retry processing shown above (0 times), perform the following setting:
TCP ULP time out value = TCP end timer value = TCP retransmission timer value
(Each timer value should be the same.)

*2 The destination existence check is the function that the E71 uses to check whether the partner remote node is operating correctly when exchange with the partner remote node that is connected by the connection has not been conducted for a set period of time. In particular, when exchange has not been conducted for a set period of time with the remote node, an echo request packet is transmitted to the partner node using a PING command (ICMP echo request/response function) to conduct an existence check of whether an echo response packet can be received. When the E71 receives a PING command echo request command, an echo response packet is automatically transmitted as a response. The E71's existence check range is restricted to remote nodes connected by the local station E71 to the Ethernet and to remote nodes opened by the router relay function.

The E71 conducts existence checks in accordance with the destination existence check settings (Refer to Item 5.4.1 [1] (b) ②) for the setting values given in [4] to [6] in this section and during opening processing.

(Example) When the setting value is for the number of retries to be 3, the E71 conducts existence checking with the timing shown in the figure below. When errors are detected, the open error detection signal turns on and stores the error code (108H) to the open error code storage area..



*3 When changing the set value of the timers, make the relation among values as follows:

Response monitor timer value ≥ TCP ULP time out value ≥ TCP end timer value ≥ TCP retransmit timer value

TCP retransmit timer value = TCP zero window timer value > IP setup timer value

When connecting to our products (E71, AJ71E71, QE71) via a line, make the same settings for both nodes.

When connecting to products of other manufacturers via a line, set each timer value so that the following relationship is satisfied, in addition to satisfying the above relationship equation.

(Monitor timer value with the external node application software) > (TCP retransmission timer value at the other node) > (TCP retransmission timer value on the E71 side)

If they are not met, there will be a great possibility of frequent communication errors such as transmission timeout.

Remarks

In the above relationship, the number of retries for transmission from E71 can be increased or decreased by changing the TCP retransmission timer value (refer to *1).

By performing the following setting, number of retries becomes 0.

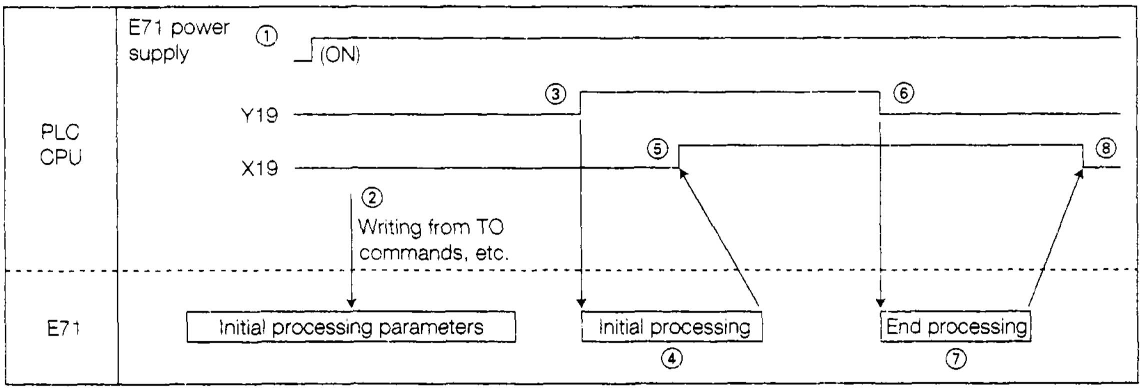
TCP ULP time out value = TCP end timer value = TCP retransmit timer value

Point

(1) It is recommended that the default values are used for each timer value. Before changing them, consult with the managers of the partner equipment and systems, then increase/decrease each set value according to the above equation.
(2) Refer to the Point in Item 3.7.2 for information regarding the parameters necessary for settings during E71 initialization processing when the E71 functions are used.

5.3.2 Initial Processing and End Processing Procedures

This section explains about the E71's initial processing and end processing procedures.



Initial processing

① Boots up the E71 installation station (turns the power on, etc.), and puts the PLC CPU in the RUN state.

② The initial processing parameters are written in the buffer memory.

* When the initial processing parameters are written, it is also all right to write the various setting values such as exchange parameters and routing information area, etc.

③ The PLC program turns the initial request signal (Y19) on.

④ The E71 executes initial processing. The initial processing results are stored in the initial processing status storage area. (Refer to Item 5.5.1)

⑤ The initial normal end signal (X19) turns on when the initial processing is normal end. When it is error end, the initial error detection signal (X1A) is turned on.

End processing (*1)

⑥ The sequence programmer turns the initial request signal (Y19) off after the next signal off is checked. (*2)

- Transmission request signal/reception end check signal (Y0 to Y7)
  Transmission normal end signal/reception end signal (X0, X2...)
  Transmission error detection signal (X1, X3...)
- Open request signal (Y8 to YF), open end signal (X10 to X17)
  Open error detection signal (X18)

⑦ The E71 executes the end processing.

⑧ The initial normal end signal (X19) turns off when the end processing is normal end. When it is error end, the initial error detection signal (X1A) turns on.

*1 This is the end processing when exchange prohibited is set by the Exchange Instruction During STOP using the E71 buffer memory (address:496). Conduct the end processing when setting exchange enable in accordance with Item 5.6.

*2 When end processing is requested when the communication line is in the connection (connection is open) state, end processing is performed after close processing is executed for the open communication line.

5.3.3 Example Program

This section explains the sequence program example for conducting E71 initial processing and end processing.

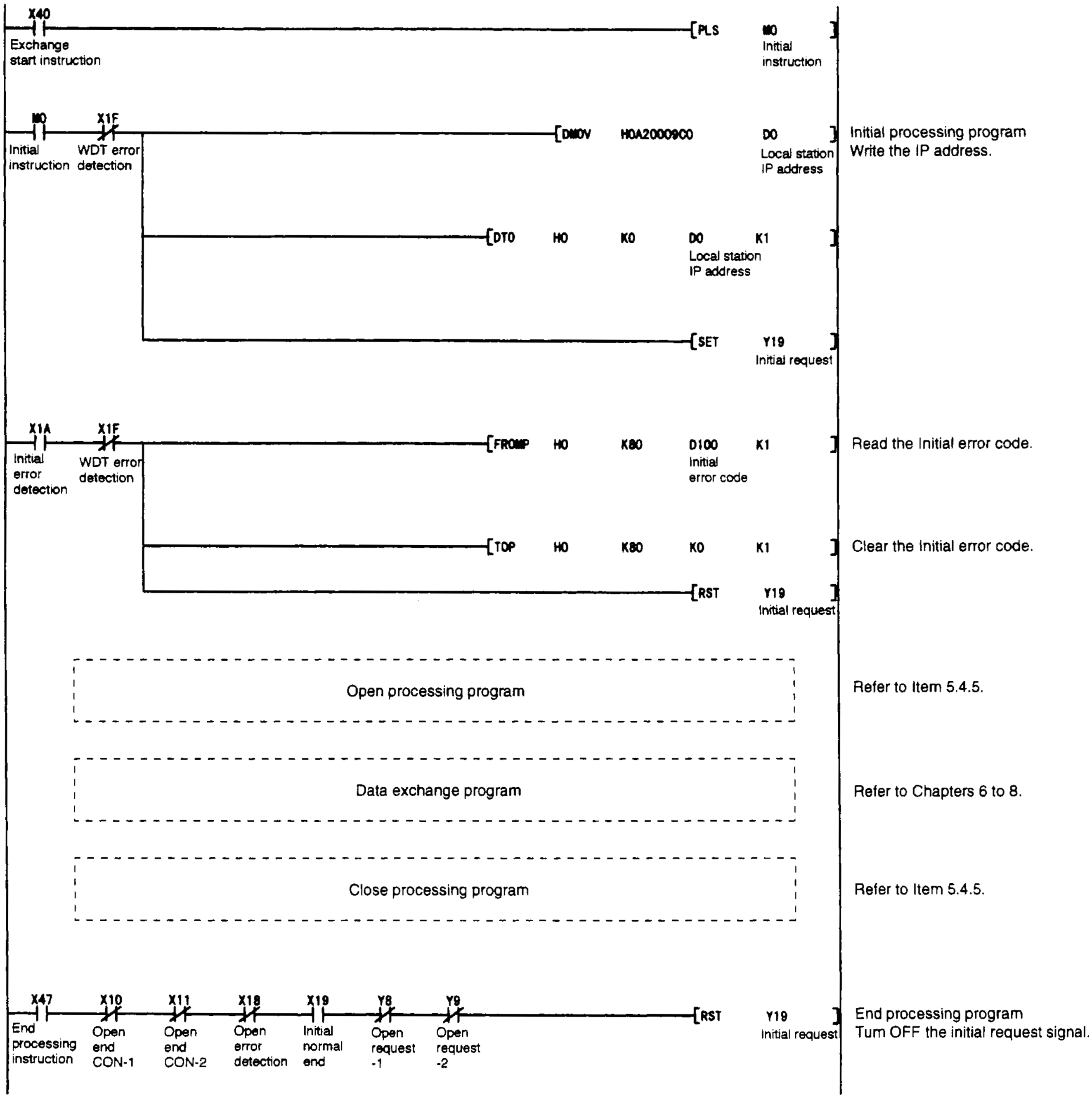
(Example) The following is an example program.

(1) The E71 is installed in the basic base's "0" slot.

(2) The initial processing parameters are shown below.

(a) The E71's IP address is "A20009C0H(162.0.9.192)."

(b) Values other than the IP address are used as default values.



5.4 Communication Line Open and Close

It is possible to exchange data at the same time with a maximum of 8 stations' remote nodes for a communication line connected by the sequence program.

The communication line can conduct fixed buffer exchange, random access buffer exchange, and reading and writing data to the PLC CPU exchange for open remote nodes. However, open processing is required when only random access buffer exchange and reading and writing data to the PLC CPU is performed.

Following is the E71's communication line open processing and close processing performed by the PLC CPU to exchange data between the E71 and remote nodes.

Remarks

(1) When the PLC program connects a communication line and uses a port to exchange data, the communication format for exchanging with the remote node can be selected during open processing using the following functions.

For each port, please specify whether TCP/IP or UDP/IP will be used for exchange.

The relationship between the E71 data exchange functions and the selectable communication formats are shown below.

Exchange Functions		TCP/IP	UDP/IP
Fixed buffer exchange	With procedure	○	○
	Without procedure	○	○
Random access buffer exchange		○	○
Reading and writing data to the PLC CPU (General data exchange)		○	○
Exchange via a router (Router relay function)		○	○

(2) Normally, in networks that use IP address, data is exchanged with remote nodes that are part of the same network (network ID is the same) as that of the local station. When data is exchanged with a remote node from a different network (different network ID) via a router, it becomes possible to exchange via a router or gateway using the PLC CPU's TCP/IP active open or UDP/IP transmission. (When exchange with a partner remote node via a router is done with the E71 in passive open, exchange can be done without using router relay functions.)

When using the router relay functions, please set the data in the buffer memory routine information area during the communication line open processing in accorandance with the explanation given in Chapter 12.

5.4.1 Data for Opening

This section explains about the exchange parameter settings area used to conduct communication line open processing. (Refer to Point at the end of Item 3.7.2 about whether the parameter should be set.)

● **Exchange Parameter Settings Area**

Buffer Memory

(Address)	Exchange Parameter Settings Area (64 Words)		Default Value
10H (16)	Connection No.1	Usage available settings area (1 word each)	0H (0)
11H (17)	Connection No.2		0H (0)
12H (18)	Connection No.3		0H (0)
13H (19)	Connection No.4		0H (0)
14H (20)	Connection No.5		0H (0)
15H (21)	Connection No.6		0H (0)
16H (22)	Connection No.7		0H (0)
17H (23)	Connection No.8		0H (0)
18H (24)	E71's Port No.	Exchange address settings area (For Connection No.1 7 words)	0H (0)
19H (25) 1AH (26)	Remote node IP address		0H (0)
1BH (27)	Remote node port No.		0H (0)
1CH (28) 1DH (29) 1EH (30)	Remote node (L) Ethernet to Address (*1) (H)		FFFFFFFFFFFFH
1F to 25H (31 to 37)	E71's port No. to	Exchange address settings area (For connection No.2 7 words)	(Same as connection No.1)
26 to 2CH (38 to 44)	E71's port No. to	Exchange address settings area (For connection No.3 7 words)	(Same as connection No.1)
2D to 33H (45 to 51)	E71's port No. to	Exchange address settings area (For connections No.4 7 words)	(Same as connection No.1)
34 to 3AH (52 to 58)	E71's port No. to	Exchange address settings area (For connections No.5 7 words)	(Same as connection No.1)
3B to 41H (59 to 65)	E71's port No. to	Exchange address settings area (For connections No.6 7 words)	(Same as connection No.1)
42 to 48H (66 to 72)	E71's port No. to	Exchange address settings area (For connections No.7 7 words)	(Same as connection No.1)
49H (73)	E71's port No.	Exchange address settings area (For connections No.8 7 words)	0H (0)
4AH (74) 4BH (75)	Remote node IP address		0H (0)
4CH (76)	Remote node port No.		0H (0)
4DH (77) 4EH (78) 4FH (79)	Remote node (L) Ethernet to Address (*1) (H)		FFFFFFFFFFFFH

*1 If the partner remote node connected by the communication line has an ARP function (broadcast), please make the default value (FFFFFFFFFFFFH).

● **Instruction Area**

Buffer Memory

(Address)	Instruction Area (1 Word)	Default Value
1F0H (496)	Communication specification during STOP area (1 word)	0H (0)

1 Usage available settings area (Default value = 0H) .... address 10H to 17H (16 to 23)

(a) Set the fixed buffer usage availability for open processing to whether or not to perform destination existence check.

(b) Conduct this setting before open processing for each connection.

(Bit position)	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	⑥		0				⑤	④	③	0					②	①
(For memo)			0	0	0	0				0	0	0	0	0		

⑥ Open method
00: Active, UDP/IP
10: Unpassive
11: Fullpassive

⑤ Fixed buffer exchange
0: With procedure
1: Without procedure

④ Communication format
0: TCP/IP
1: UDP/IP

③ Pairing open
0: No pairs
1: Pairs

② Existence check
0: No check
1: Check

① Fixed buffer usage
0: For transmission/does not communicate
1: For reception

① Fixed buffer usage availability setting (b0)

- When conducting exchange using a fixed buffer, set whether the fixed buffer will be used for transmission or reception for the particular connection.
- When conducting transmission and reception using one specific node and fixed buffer, two fixed buffers are required for transmission and reception, so please set two connections.
- Specify one of the following setting values.

  0 : For transmission or not to perform fixed buffer exchange (default value)

  1 : For reception
- From remote node, random access buffer exchange and reading and writing data to the PLC CPU exchange can be conducted by either reception setting or transmission setting for usage of fixed buffer.

② Destination existence check setting (b1)

- Set the E71 to check whether the partner remote node is operating normally when exchange with the partner remote node for its connection open processing as ended has not been conducted for a specific period of time. (*2 Refer to Item 5.3.1)
- Specify one of the following specification values.

  0 : Does not check existence (no check) (default value)

  1 : Checks existence (has check)

  * When existence check is selected, the E71 conducts an existence check for the destination at each specified time interval to check whether the connection destination (partner destination) is operating correctly. The E71 will conduct the following process if an error occurs during the existence check.

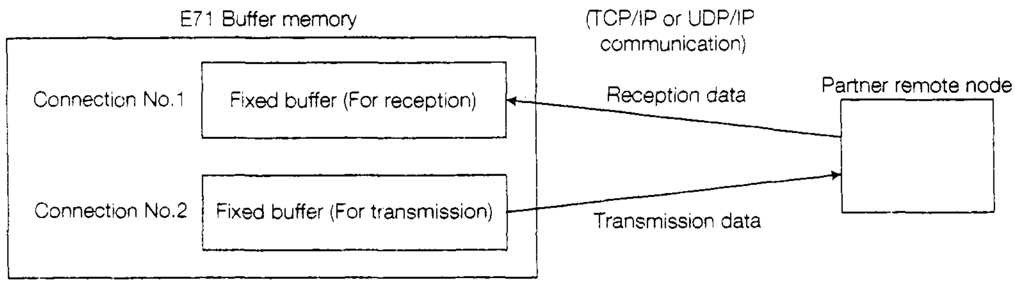
  - Force closes the line and stores the error information in the buffer memory error log area (address 169 to 179).
  - The open error detection signal (X18) is turned on when the open end signal (X10 to X17) is turned off.
- When changing the exchange partner remote node during the middle of an operation at the UDP/IP connection, make the setting 0 (no existence check).

  When 1 (check existence) is set, the E71 conducts an existence check on the first exchange partner after UDP/IP open. An existence check is not performed for exchange partners after the change.

③ Pairing open setting (b7)

- Sets whether one of the partner remote nodes' ports is connected when the E71 resumption connection and transmission connection are made into one pair when fixed buffer exchange (either with procedure or without procedure can be selected) is conducted.

(Example)



- When pair opening is set, the subject connection No.'s fixed buffer and the next connection No.'s fixed buffer are paired. (When the connection No. to be opened is "8," the connection No.8's fixed buffer and the connection No.1's fixed buffer are paired.)
- When pairing is set the next connection No. side usage application setting, exchange parameter setting, and open processing (open request signal is on) are not necessary. (E71 conducts automatically.)

  The open end signal for the next connection No. is turned on by the open processing for the connection No. that sets the pairing open.
- The fixed buffer usage application for the next connection No. side is determined by the fixed buffer usage application setting (b0) of the connection No. side that sets the pairing open.

	Setting buffer usage application	
Connection No. side that sets the pairing open	For reception use	For transmission use
Next connection No. side	For transmission	For reception

- The open processing image when pairing is set is shown in Item 5.4.4.
- Set one of the following pairing open setting (b7) setting values.

  0 : Pairing open not conducted (default value)

  1 : Pairing open conducted

④ Communication format (Protocol) settings (b8)

- Sets whether TCP/IP or UDP/IP is used as the communication protocol for each connection.
- Select one of the following setting values.

  0 : TCP/IP (default value)

  1 : UDP/IP

⑤ Fixed buffer exchange procedure existence setting (b9)

- Sets the exchange method used to conduct fixed buffer exchange.
- Specify one of the following setting values.

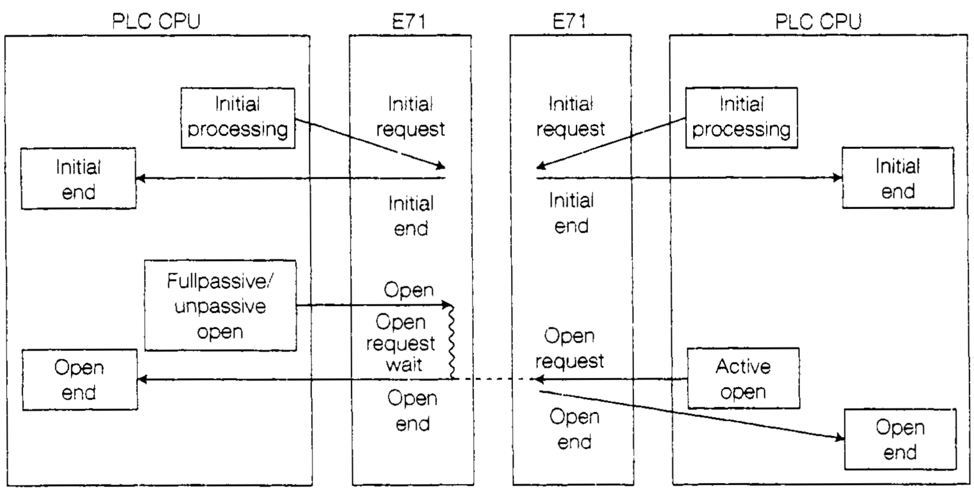
  0 : With procedure (default value)

  1 : Without procedure

  * When with procedure is selected, in the subject connection, fixed buffer exchange with procedure, random access buffer exchange, and reading and writing data to the PLC CPU exchange can be conducted.
  When without procedure is selected the subject connection becomes a without procedure fixed buffer exchange special use, so fixed buffer exchange with procedure, random access buffer exchange, and reading and writing data to the PLC CPU exchange cannot be conducted at the same time as exchange without procedure.

⑥ Open method setting (b14, b15)

- This setting is valid only when the communication format (protocol) is TCP/IP.
- Setting is not required when UDP/IP is the communication method, so make the setting "00."
- When opening using TCP/IP, open the active open node after the Full passive/ Unpassive open node open processing end.



- Specify one of the following settings.

  00 : Active open or UDP/IP (default value)

  10 : Unpassive open

  11 : Full passive open

Remarks

Shows the differences between each open format.

① Active open method

Conducts active open processing for the remote nodes that are in the TCP connection open passive state (Full passive/Unpassive open state).

② Full passive open format

Conducts TCP connection passive open processing for only the specified nodes that are set in the exchange address setting area. Changes to the active open request wait state from the remote node that is set in exchange address setting area.

③ Unpassive open format

Conducts TCP connection passive open processing for all the remote nodes connected to the network. All the remote nodes in the network change to the active open request wait state.

(c) Following is an example of the usage available setting area data setting.

(Bit position)	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
	⑥		0				⑤	④	③	0					②	①

① Fixed buffer usage availability
② Destination existence check
③ Pairing open setting
④ Communication format
⑤ Fixed buffer communication procedure existence
⑥ Open method setting

(Example 1) When set to ③ is "0" (Pairing open not done), ⑤ is "0" (With procedure).

		④ : 0(TCP) ⑥ : 00 (Active)	④ : 0(TCP) ⑥ : 10 (Unpassive)	④ : 0(TCP) ⑥ : 11 (Fullpassive)	④ : 1 (UDP)
① : 0 (For transmission)	② : 0 (Does not check)	0000H	8000H	C000H	0100H
	② : 1 (Checks)	0002H	8002H	C002H	0102H
① : 1 (For reception)	② : 0 (Does not check)	0001H	8001H	C001H	0101H
	② : 1 (Checks)	0003H	8003H	C003H	0103H

(Example 2) When ③ is "0" (Pairing open not done), ⑤ is "1" (Without procedure).

		④ : 0(TCP) ⑥ : 00 (Active)	④ : 0(TCP) ⑥ : 10 (Unpassive)	④ : 0(TCP) ⑥ : 11 (Fullpassive)	④ : 1 (UDP)
① : 0 (For transmission)	② : 0 (Does not check)	0200H	8200H	C200H	0300H
	② : 1 (Checks)	0202H	8202H	C202H	0302H
① : 1 (For reception)	② : 0 (Does not check)	0201H	8201H	C201H	0301H
	② : 1 (Checks)	0203H	8203H	C203H	0303H

(Example 3) ③ is "1" (Pairing open is done), ⑤ is "0" (With procedure) is set.

		④ : 0(TCP) ⑥ : 00 (Active)	④ : 0(TCP) ⑥ : 10 (Unpassive)	④ : 0(TCP) ⑥ : 11 (Fullpassive)	④ : 1 (UDP)
① : 0 (For transmission)	② : 0 (Does not check)	0080H	8080H	C080H	0180H
	② : 1 (Checks)	0082H	8082H	C082H	0182H
① : 1 (For reception)	② : 0 (Does not check)	0081H	8081H	C081H	0181H
	② : 1 (Checks)	0083H	8083H	C083H	0183H

2 **Exchange address setting area**

(a) Sets the local station E71's port No. partner remote node IP address, port No., etc., when communication line is connected using open processing.

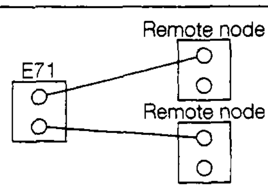
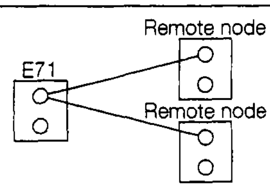
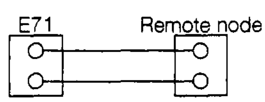
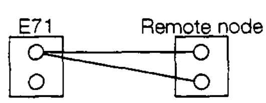
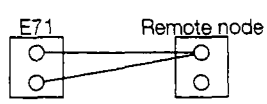
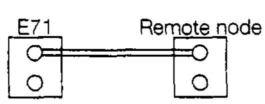
When setting the pairing open using the usage application setting bit7 (b7), the next connection No. side exchange parameter setting is not required. (This is automatically conducted by the E71.) The next connection No. open end signal is turned on by the open processing for the connection No. that sets the pairing open.

(b) For the setting values, please specify the values set by the network manager.

(c) Sets the data in accordance with the contents of the usage availability setting area's open procedure settings (b14, b15) shown in 1 when settings are conducted for each connection. (refer to key points in Item 3.7.2) Sets these settings before open processing is conducted during TCP open, open processing is conducted before UDP open processing, and before data transmission and reception.

① E71's port No. setting (Default value = 0H): Address 18H (24...)

- Sets the local station E71's port No.
- The setting values are specified to between 100H and FFFEH. As far as possible, it is recommended that a port No. be set at 401H or later. Set to No. that is not being used elsewhere.
- Following are the precaution items for port Nos. when multiple connections are made between remote nodes and the local station using open processing. (In the diagram, the nodes are denoted by a square, and the port Nos. are denoted by the circles.)

Connection State (Shows O: Port (Port No.))	Connection Description	Communication Protocol	
		TCP	UDP
	Also sets multiple local station port Nos. even though connections are made with multiple nodes.	○	○
	Sets a single local station port No. when connections are made with multiple nodes. (However, several connections must be opened.) Do not perform this when the local station is unpassive.	○	×
	Also sets multiple E71 port Nos. even though connections are made with multiple remote node ports.	○	○
	Sets a single E71 port No. even though connections are made with multiple remote nodes. (However, several connections must be open.) Do not perform this when the local station is unpassive.	○	×
	Sets multiple E71 port Nos. even though connection is made to the same remote node port. (However, several connections must be open.)	○	○
	Multiple settings when the remote node same port and the E71's same port is only possible for pairing open settings.	○	○

② Remote node IP address setting (Default value = 0H)
....Address 19H to 1AH....(25 to 26...)

- Sets the IP address for the partner remote node to which exchange will be conducted.
- Specify the settings value as other than 0H and FFFFFFFFH except for when simultaneous broadcast communication is performed exchanged without procedure (UDP/IP) by fixed buffer. (FFFFFFFFH is the setting value for the simultaneous broadcast communication mentioned above.) Conduct setting after checking the mutual remote node's IP address.

③ Remote node port No. setting (Default value = 0H) .... Address 1BH....(27...)

- Sets the port No. for the primary remote node for which exchange will be done.
- Specify the settings value as between 100H and FFFEH except when simultaneous broadcast communication is performed with exchange without procedure (UDP/IP) by the fixed buffer. As far as possible, it is recommended that a port No. be set at 401H or later. (FFFFH is the setting value for the simultaneous broadcast communication described above.) Before setting, check the partner remote node's port No.

④ Remote node ethernet address setting (Default value = FFFFFFFFFFFFH)
.... Address 1CH to 1EH....(28 to 30...)

- When the partner remote node to which exchange is being conducted does not have ARP functions, set the partner remote node's Ethernet address.
- Set the settings values to those shown below.

When the partner remote node has ARP functions .... 0H or FFFFFFFFFFFFH

When the partner remote node does not have ARP functions
.... Partner remote node's Ethernet address (except 0H and FFFFFFFFFFFFH)

When specifying other than 0H and FFFFFFFFFFFFH, check the partner remote node's Ethernet address before making the settings.

* When this setting value is 0H or FFFFFFFFFFFFH, the E71 conducts processing as if the partner remote node has ARP functions.

(Example) The settings data when the Ethernet address is 080070220004H is shown below. (For connection No.1)

Address	Buffer Memory
2CH(44)	0004H
2DH(45)	7022H
2EH(46)	0800H

⇐ Write ⇐ 0800H 7022H 0004H
D117 D116 D115

Point

(1) Determine the setting values by consulting with the partner equipment and the system's managers.
(2) For information regarding the parameters required for setting when E71 open processing is conducted when E71 functions are used, refer to Point in Item 3.7.2.

3 Exchange instruction area during STOP (default value = 0H)

.... **Address 1F0H(496)**

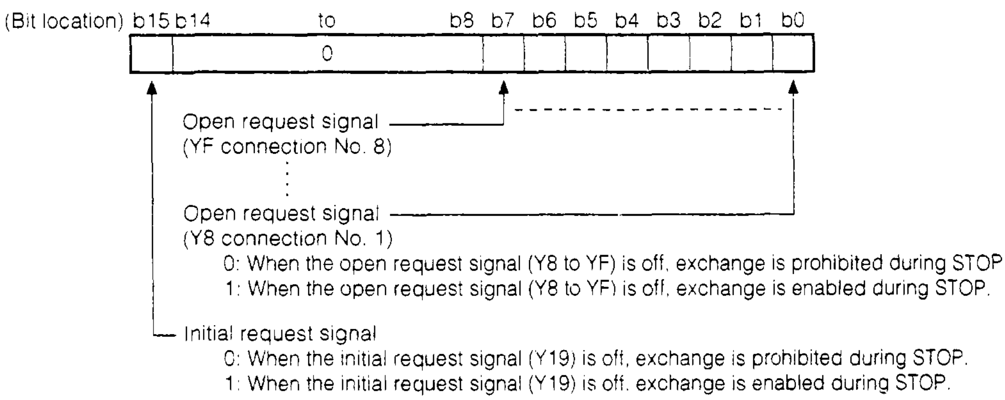
(a) This is the setting that is used to continue the next data exchange for the E71 from the next remote nodes even when the PLC CPU of the station installed in E71 is in the STOP status and the E71 open request signals (Y8 to YF) and initial request signal (Y19) are off.

- Random access buffer exchange
- Exchange of read/write data in the PLC CPU

(b) This specification is conducted using the communication line unit, and the setting value can be changed even after communication line open processing. Use the system specifications to set the exchange enable/prohibit during STOP.

(c) Conduct the setting to this specification area, continuing data exchange, and reopen processing in accordance with Item 5.6.

(d) The setting value is specified as follows.



(Example setting)

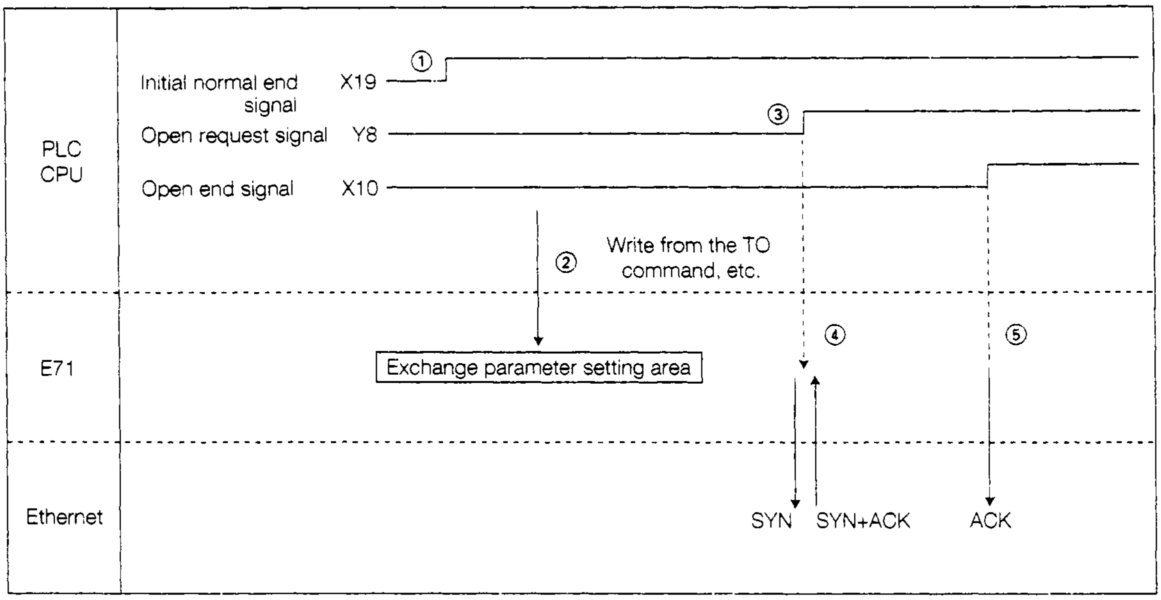
- Set to 0003H to continue exchange between connection No. 1 and connection No. 2 when the initial request signal (Y19) is left on and after the open request signals (Y8, Y9) are turned from on to off.
- Set to 8003H to continue exchange between connection No. 1 and connection No. 2 after the initial request signal (Y19) and the open request signals (Y8, Y9) are turned from on to off.

5.4.2 Communication Line Open Processing Procedure

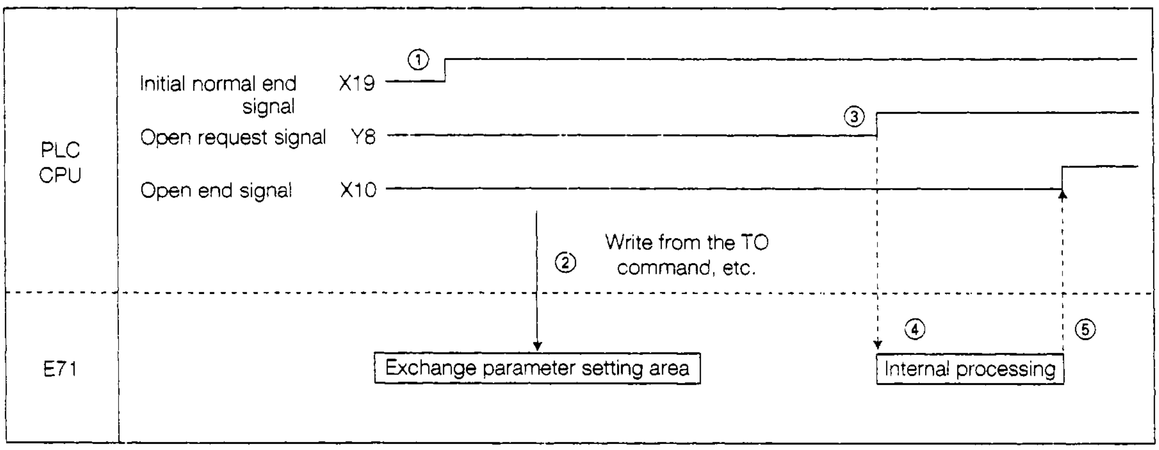
This section explains the open processing procedure for connecting a communication line from the E71 to a remote node using an example for connection No.1.

To conduct open processing, initial processing must be completed.

1 Open processing procedure using TCP



2 Open processing procedure using UDP



① Initial processing is conducted by the initial request signal (Y19). Initial normal end signal (X19) turns on. (Refer to Item 5.3)

② The TO command, etc., causes the sequence program to write the setting values (parameters) in the buffer memory exchange parameter settings area.

③ The open request signal (Y8) is turned on by the sequence program.

④ The E71 executes the open processing. (*1)

(For TCP)
For active open .... Open request (SYN) is transmitted.
For passive open .... An open request from the partner remote node is waited.

(For UDP)
Executes internal processing.

⑤ The open end signal (X10) is turned on when the open processing/internal processing is normal end.

The open error code is stored in the buffer memory, and the open error detection signal (X18) is turned on when the open processing/internal processing is error end. (The open end signal (X10) does not turn on.) (*2)

When the open request signal is turned OFF while an open error is being generated, the open error detection signal will be turned OFF if no open errors are generated in other line connections.
(If the open request signal (Y8 to YF) is off for all connections in which an open error is currently occurring, the open error detection signal (X18) is turned off.)

*1 When the initial request signal (Y19) is off or the open request signal (Y8) is turned off during open processing, closed processing and end processing are performed after the open processing end.

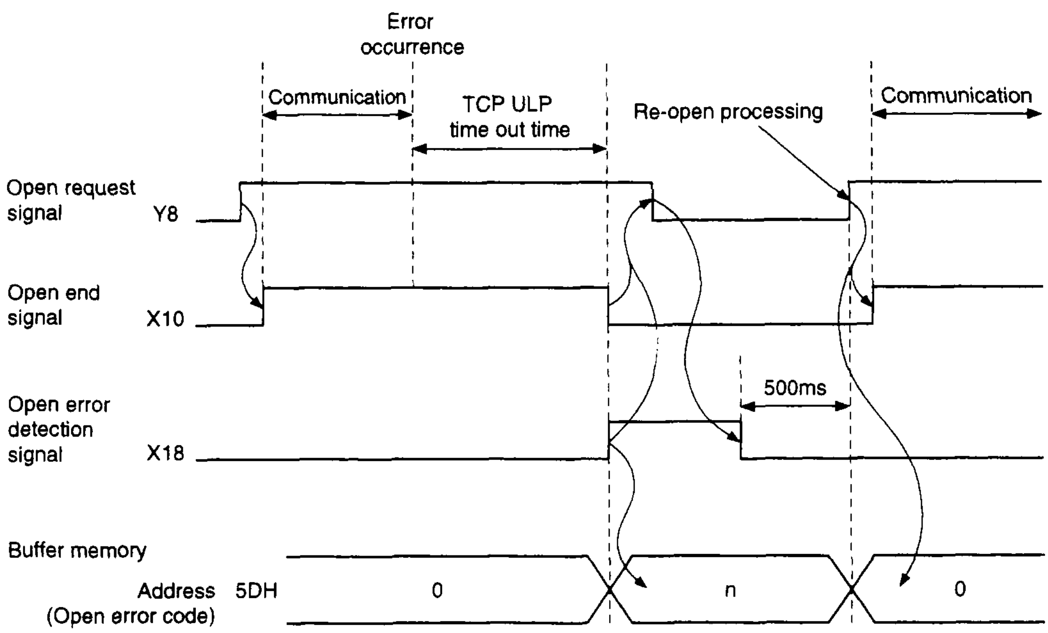
*2 The open state and error codes during error end are checked by the next buffer memory.

- The exchange state storage areas (information storage area by connection: address 89 to 168) open error code area.
- Error log area (address 169 to 179).

The error code stored in the open error code area will be cleared (n → 0) when the open request signal is turned ON again.

Remarks

The following is the timing chart of the reopen processing when the open end signal is turned off due to an error in the TCP/IP communication. (When connection No. 1)



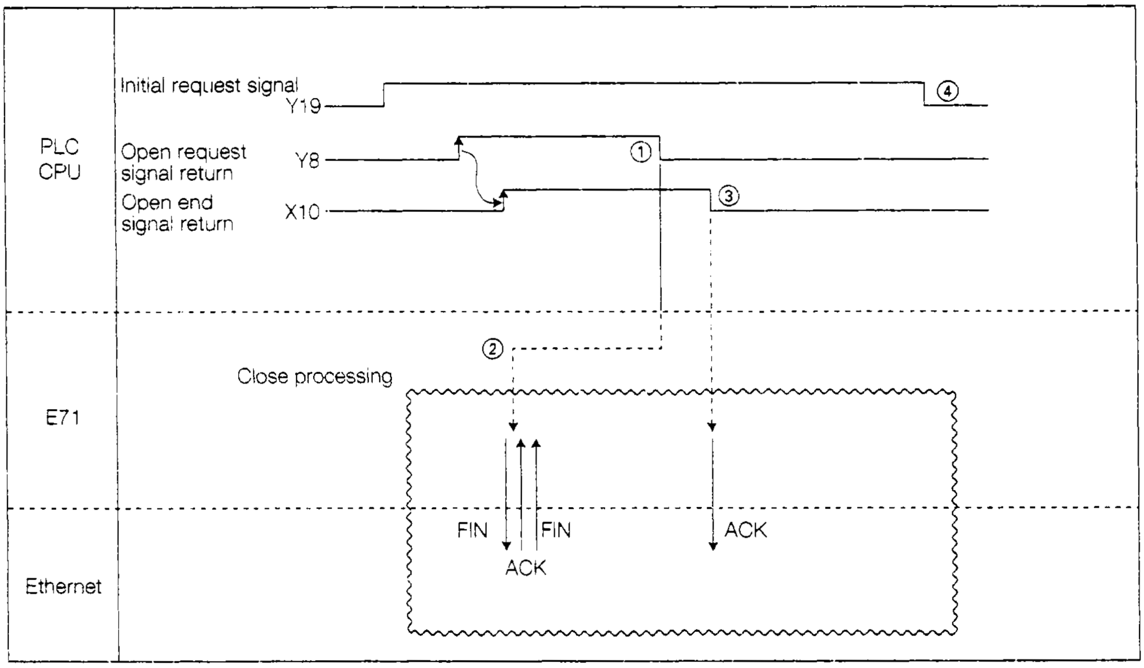
5.4.3 Communication Line Close Processing Procedure

This section explains the close processing for closing (disconnecting) the communication line that was connected between the E71 and the remote node by open processing using an example for connection No.1.

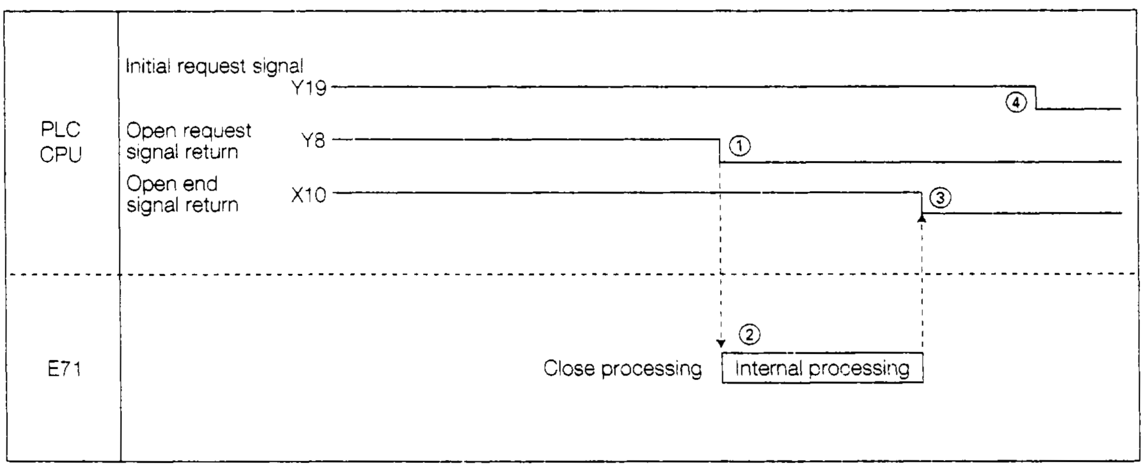
The close processing timing must be conducted by making arrangements with the partner remote node.

1 For closing from the E71 end

(a) Close processing procedures for TCP



(b) Close processing procedures for UDP



① The open request signal (Y8) is turned off by the sequence program.

② The E71 executes closed processing.

* If the transmission request signal/reception completion confirmation signal (Y0, Y1...) for the fixed buffer communication of the corresponding connection is off when the open request signal (Y8, Y9...) is turned off, the E71 turns off the corresponding input signal (X) listed below:

  - Transmission normal completion signal/reception completion signal (X0, X2...)
  - Transmission error detection signal/reception error detection signal (X1, X3...)

  (Example: For connection 1)

  If the transmission request signal/reception completion confirmation signal (Y0) for connection 1 is off when the open request signal (Y8) is turned off, the E71 turns off the following input signals (X).

  - Transmission normal completion signal/reception completion signal (X0)
  - Transmission error detection signal/reception error detection signal (X1)

③ When close processing ends the open end signal (X10) turns off for either normal close or error close. (The open request signal (Y8) can be turned on immediately after the open end signal (X10) turns off.)

④ Initial request signal (Y19) is turned off by the sequence program.

Point

The open end signal (X10 to X17) is automatically turned off and the communication line will be closed in the following cases in addition to a close request. To reopen, first turn off the open request signal (Y8 to YF) once, and then conduct open processing.

(1) When the DIP switch SW1 is set to off (the line is closed by a TCP/ULP timeout error), the open end signal will turn off when the following timeout occurs.
   ① Timeout during TCP transmission
   ② Partner remote node existence check function timeout
   ③ At this time the open error detection signal (X18) turns on. (*1)
      * When the DIP switch SW1 is set to on (Line does not close), the line will not close even when the above timeout occurs. (The open error detection signal (X18) will not turn on.) However, the error code will be stored in the buffer memory described in *1 below.

(2) The open end signal will turn off when a close or ABORT (RST) command is received from the partner remote node.

(3) The open end signal will turn off when transmission of the ABORT command is conducted. At this time, the open error detection signal (X18) will turn on. (*1)

(4) When the active open request is received again from the other node during the TCP/IP open end state, the open end signal turns OFF after the RST command is transmitted.

(5) The open end signal will turn off when the E71's abort command is transmitted including that described above (Refer to Item 3.5.3).

*1 There error code during the open status for the error end can be checked using the buffer memory exchange status storage (addresses 89 to 168) and error log area (addresses 169 to 179).

- Error during open processing : Open error code area
- Error during data exchange

  When conducting transmission using the fixed buffer:
  Set buffer transmission error code area

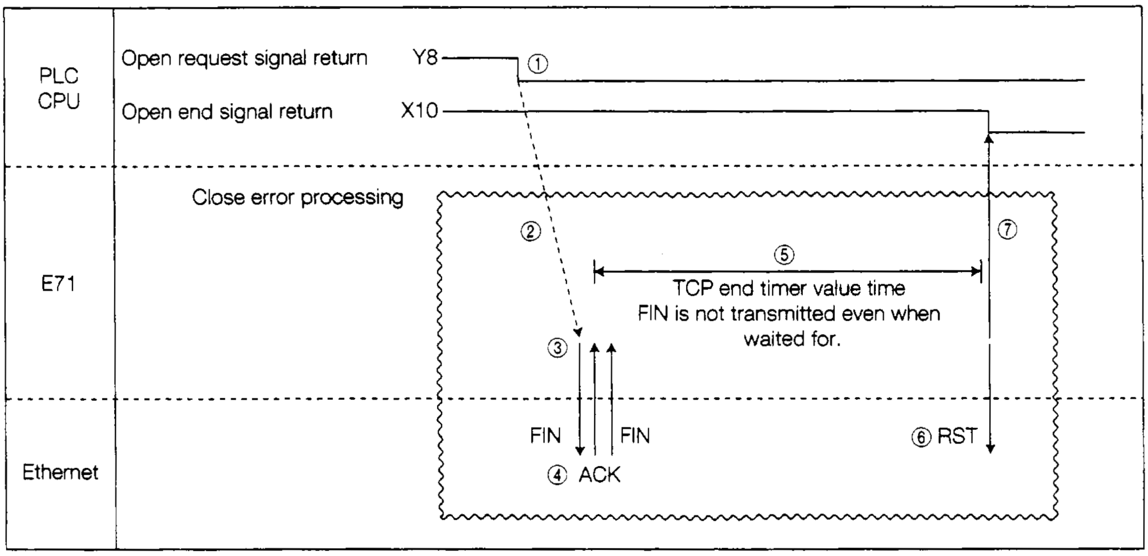
  When conducting transmission using any other than the fixed buffer :
  Error log area

The error code stored in the open error code area will be cleared (n → 0) when the open request signal is turned ON again.

(c) Processing (TCP) when error end is done by close

Normally, when closed by the E71, the E71 transmits an FIN and then ACK, FIN are returned from the partner remote node. However, when ACK, FIN are not returned because of a partner remote node error, the E71 forcefully disconnects the connection (transmission of ABORT (RST) command).

Following is an explanation that uses an example of processing for connection No. 1.



① The open request signal (Y8) is turned off by the sequence program.

② The E71 begins close processing.

③ The E71 transmits FIN to the partner remote node.

④ The partner remote node returns ACK, FIN in response to the FIN sent by the QE71.
(If it is not returned, the E71 retransmits the FIN.)

⑤ The E71 waits for ACK, FIN to be transmitted by the partner remote node.
(The wait time is the TCP end value timer time.)
At this time, if ACK, FIN is transmitted, an ACK will be returned as normal processing.

⑥ If the ACK, FIN is not transmitted within the TCP end timer value time, ABORT (RST) command is transmitted to the partner remote node.

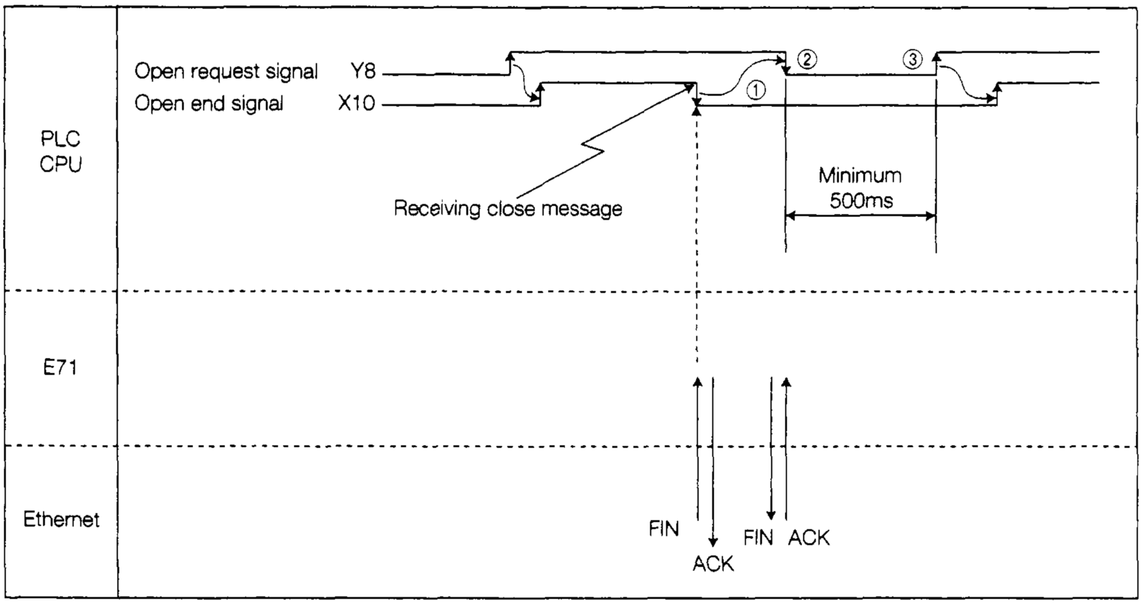
⑦ The E71 determines that close processing has ended regardless of the state of the partner remote node, and turns the open end signal (X10) to off.

Remarks

(1) When the above processing is conducted, the E71 determines that the partner remote node processing was conducted normally, so the close results are not stored in the error log area.

(2) The above processing is a unique function of the E71, and is not part of the general TCP/IP protocol.

2 **When closing from the partner remote node end**



① The open end signal (X10) is turned off when the close/ABORT (RST) command is received from the partner remote node.

② The open request signal (Y8) is turned off by the sequence program when the open end signal is turned off.

③ When reopening the open request signal (Y8) is turned on by the sequenc program after a minimum of 500ms.

Point

For the sequence program to recognize the open end from the remote node side the open end signal (X10 to X17) on time must be longer than the PLC CPU's scan time. Even if there is an open end, if a close message is received that is shorter than the ACPU scan time, the sequence program may not recognize the open end.

Remarks

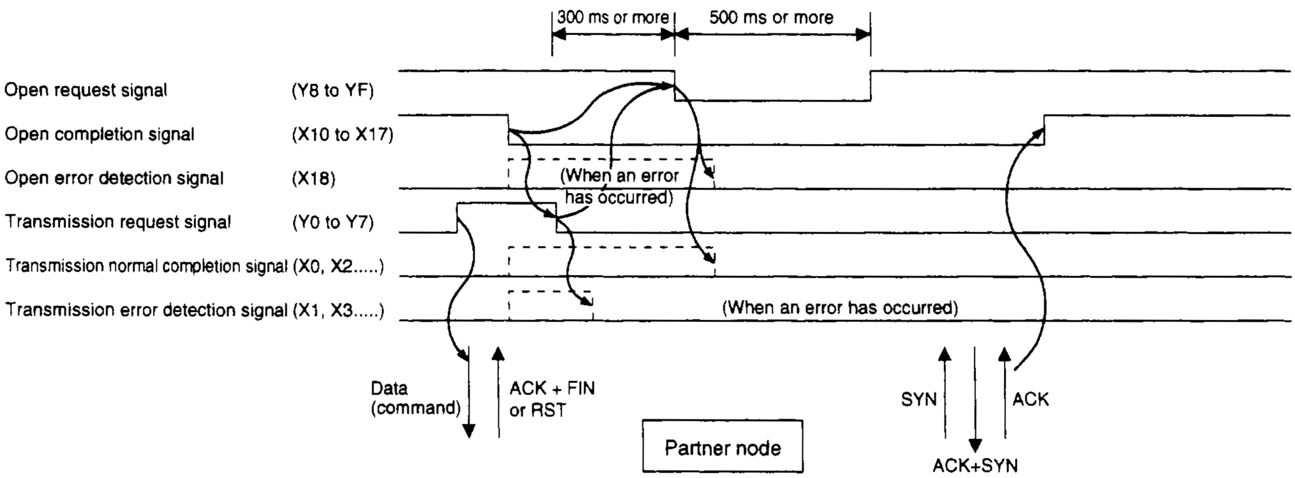
The E71 performs close processing even if the following signals for the corresponding connection are on when the reception of Close/Abort (RST) has occurred from the partner remote node prior to the completion of data communication performed immediately before.

- Transmission request signal/reception completion confirmation signal (Y0 to Y8)
- Transmission normal completion signal/reception completion signal (X0, X2...)
- Transmission error detection signal/reception error detection signal (X1, X3...)

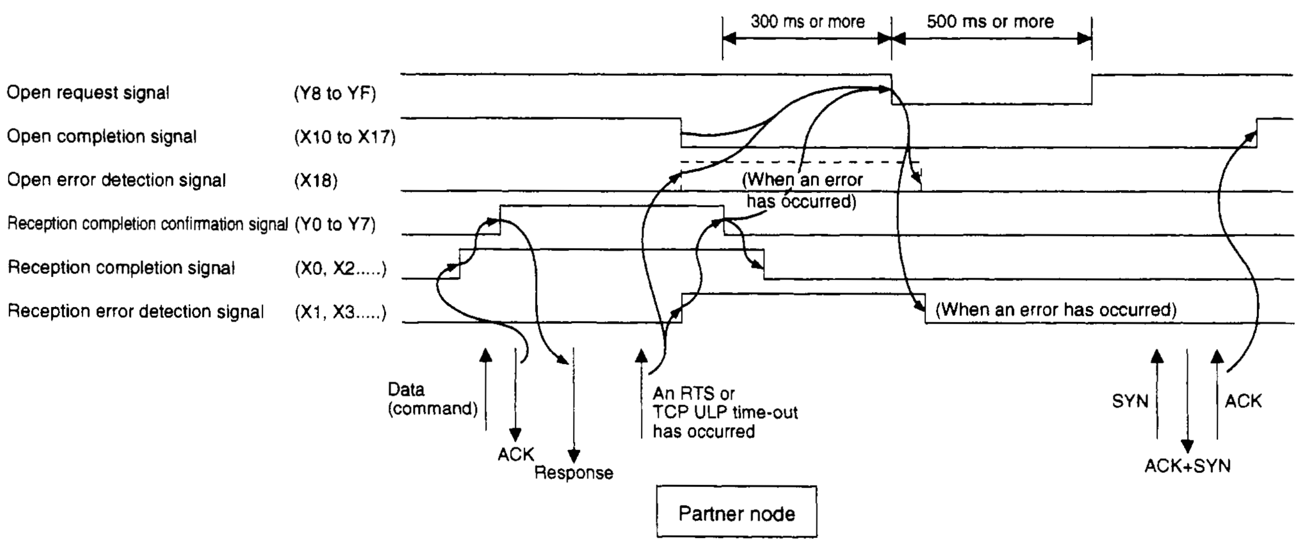
If close processing is performed for the corresponding connection while the above signals are on, turn off the open request signal (Y8 to YF) after turning off the transmission request signal/reception completion confirmation signal at the timing shown below. The E71 turns off the above input signals of the corresponding connection.

* Program examples are shown in Item 7.2 of Appendix.
* If pairing open has been performed for the corresponding connection, the signal that was specified when opening will be the target open request signal for the input/output signals shown in the figure below.

For transmission (when closed by the partner node)



For reception (when closed by the partner node)

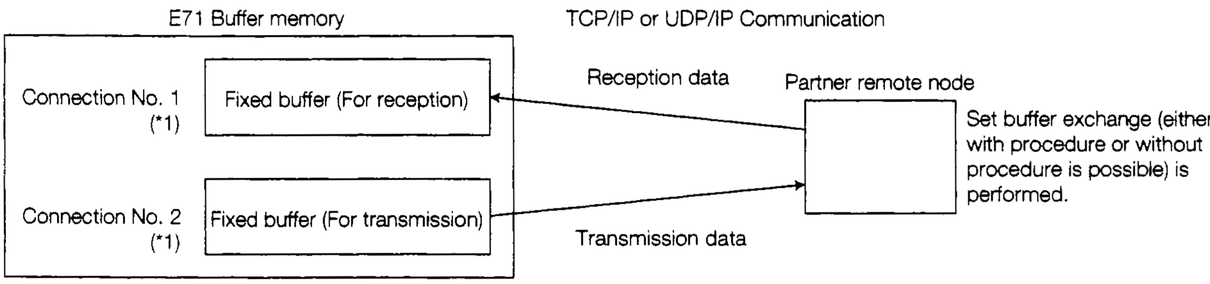


5.4.4 Pairing Open Communication Line Open Processing and Close Processing Procedures

This section explains the open processing and close processing procedures when connecting a communication line to one partner remote node port when the E71's reception connection and transmission connections are made into one pair.

Fixed buffer exchange (either with procedure or without procedure is possible) is performed by the connection which the pairing open was processed.

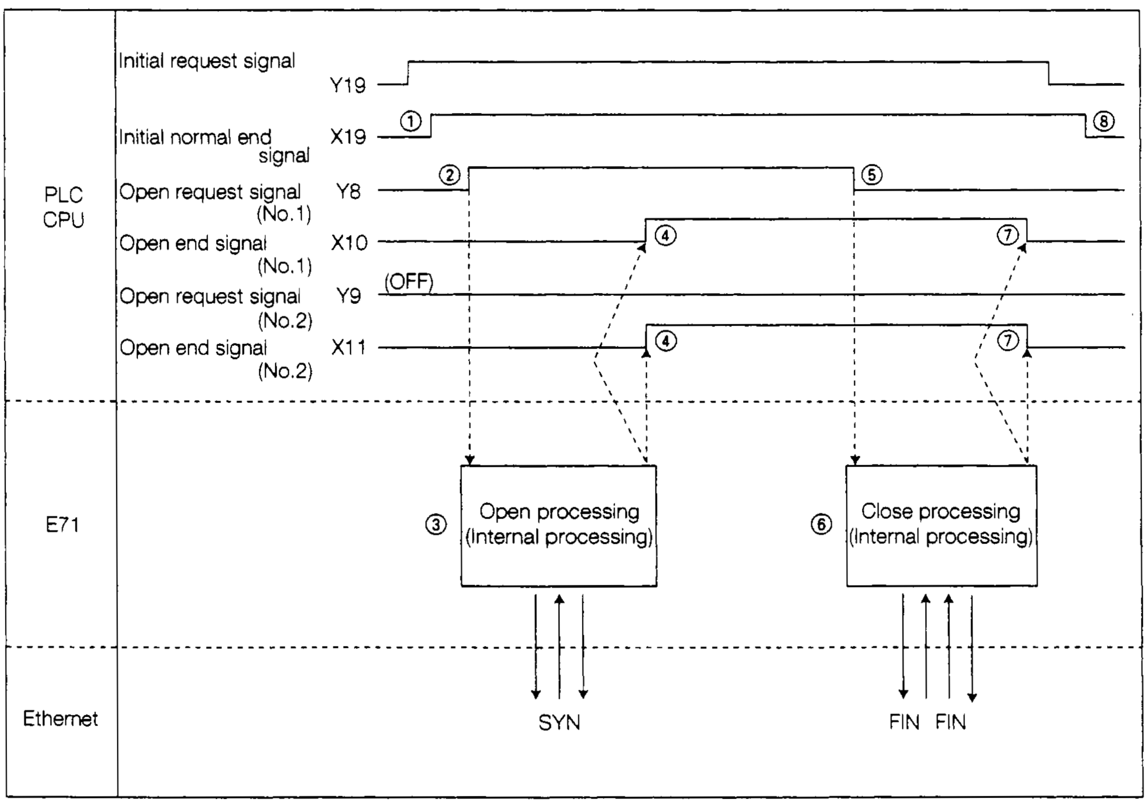
(Example)



*1 Connection No.1 and No.2's exchange parameter settings (Address 10H to 11H)

- Connection No.1 exchange parameter setting (Address 10H) : 0081H
- Connection No.2 exchange parameter setting (Address 11H) : 0000H

(When the Pairing is Connection No. 1 and Connection No. 2)



Open processing

① The initial normal end signal (X19) is turned on by the initial processing normal end which is produced when the initial request signal (Y19) is turned on. (Refer to Item 5.3)

② The setting values (parameters) are written to the buffer memory exchange parameter settings area by the sequence program in response to a TO command, etc., which turns on the open request signal (Y8).

③ The E71 executes open processing for connection No.1 and connection No.2. (*1)

(For TCP)

For active open: .... Transmits open request (SYN).

For passive open: .... Waits for an open request from the partner remote node.

(For UDP)

Internal processing is executed.

④ Open end signal (X10, X11) is turned on when the open processing/internal processing is normal end.

The open error code is stored in the buffer memory, and the open error detection signal (X18) is turned on when the open processing/internal processing is error end.

(The open end signal (X10, X11) is not turned on.) Refer to Item 5.4.2 *2

If the open request signal is turned off when an open error occurs, the open error detection signal turns off unless an open error has occurred in other line connection.
(If the open request signal (Y8 to YF) for all connections in which an open error is currently occurring is off, the open error detection signal (X18) is turned off.)

*1 When the initial request signal (Y19) is off or the open request signal (Y8) is turned off during open processing, closed processing and end processing are performed after the open processing end.

Close processing

⑤ The open request signal (Y8) is turned off by the sequence program.

⑥ The E71 executes close processing for connection No.1 and connection No.2.

⑦ The open end signal (X10, X11) is turned off when the close processing ends.

⑧ The initial request signal (Y19) is turned off by the sequence program.

Point

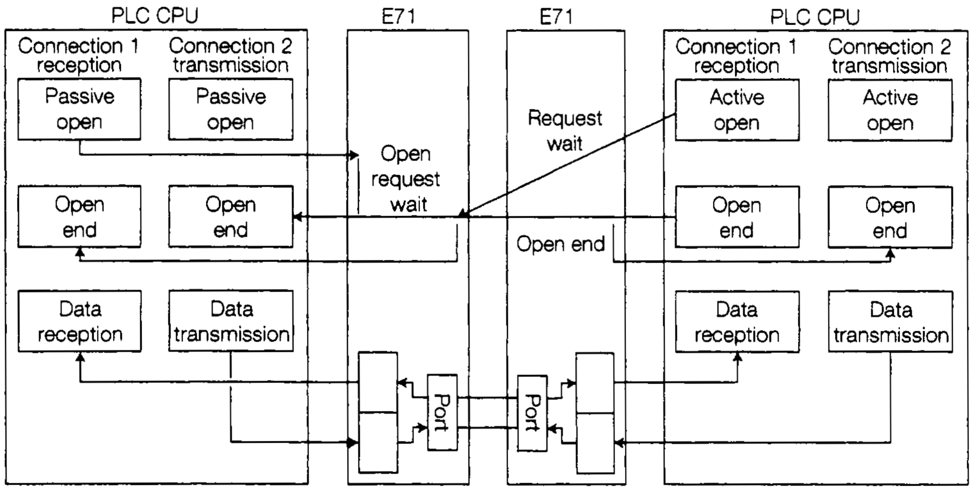
(1) An exchange parameter setting to correspond to the next connection No. of the connection No. that was pairing opened is not required. (Ignored.)
For information regarding the exchange parameter setting (usage application setting and exchange address setting) refer to Item 5.4.1 1 ③.

(2) This is the remote node that is connected by the remote node in the Ethernet that is connected by the E71 or by the router relay function (Refer to Chapter 12) that sets the remote node range with which exchange can be done using pairing open.

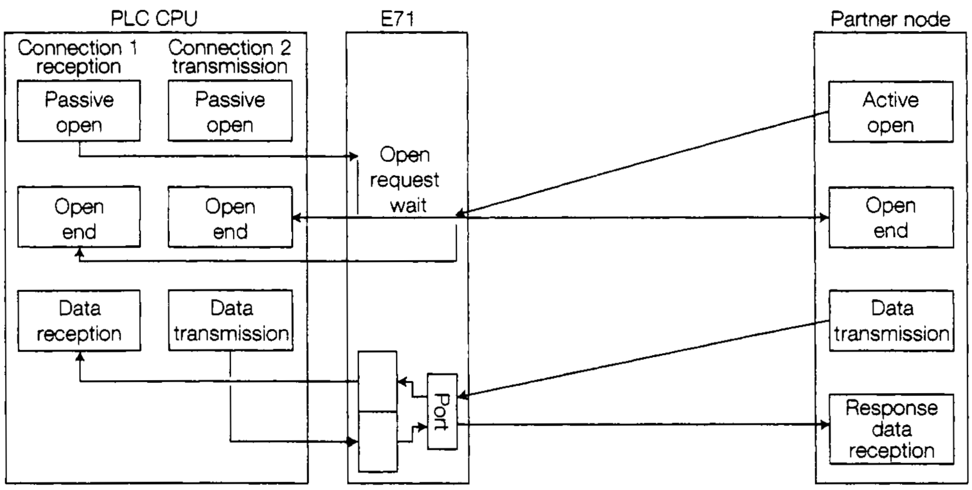
Remarks

(1) Following is shown the open processing image when pairing is set.

<Example 1> Pairing between E71s

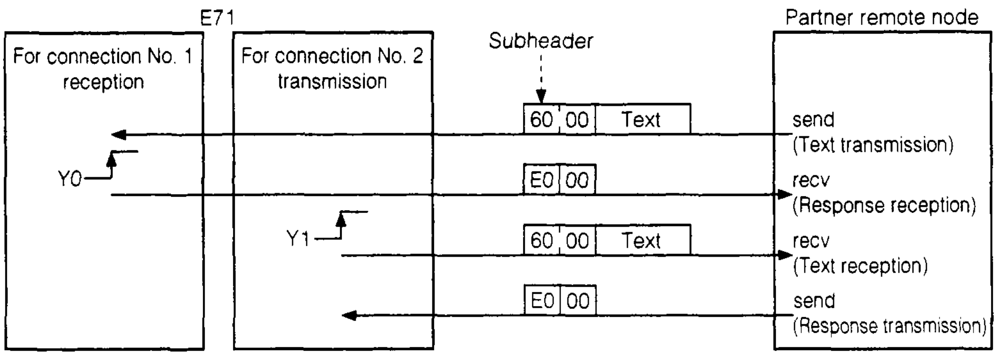


<Example 2> Connection between E71 and partner node



(2) When conducting fixed buffer exchange using pairing open, use one port each for the E71 and the partner remote node. The E71 uses the receive data subheader to determine whether the reception data from the remote node is text or response. The remote node also uses the subheader to determine whether the received data is text or response.

(Example) For fixed buffer exchange (with procedure)



(3) For more details on cases when the reception of Close/Abort (RST) has occurred from the partner remote node prior to the completion of the data communication performed immediately before, refer to Remarks in [2] of Item 5.4.3.

5.4.5 Example Program

This section explains an example sequence program used to do the connection open processing for the E71 and a remote node.

(Example) The following is an example program.

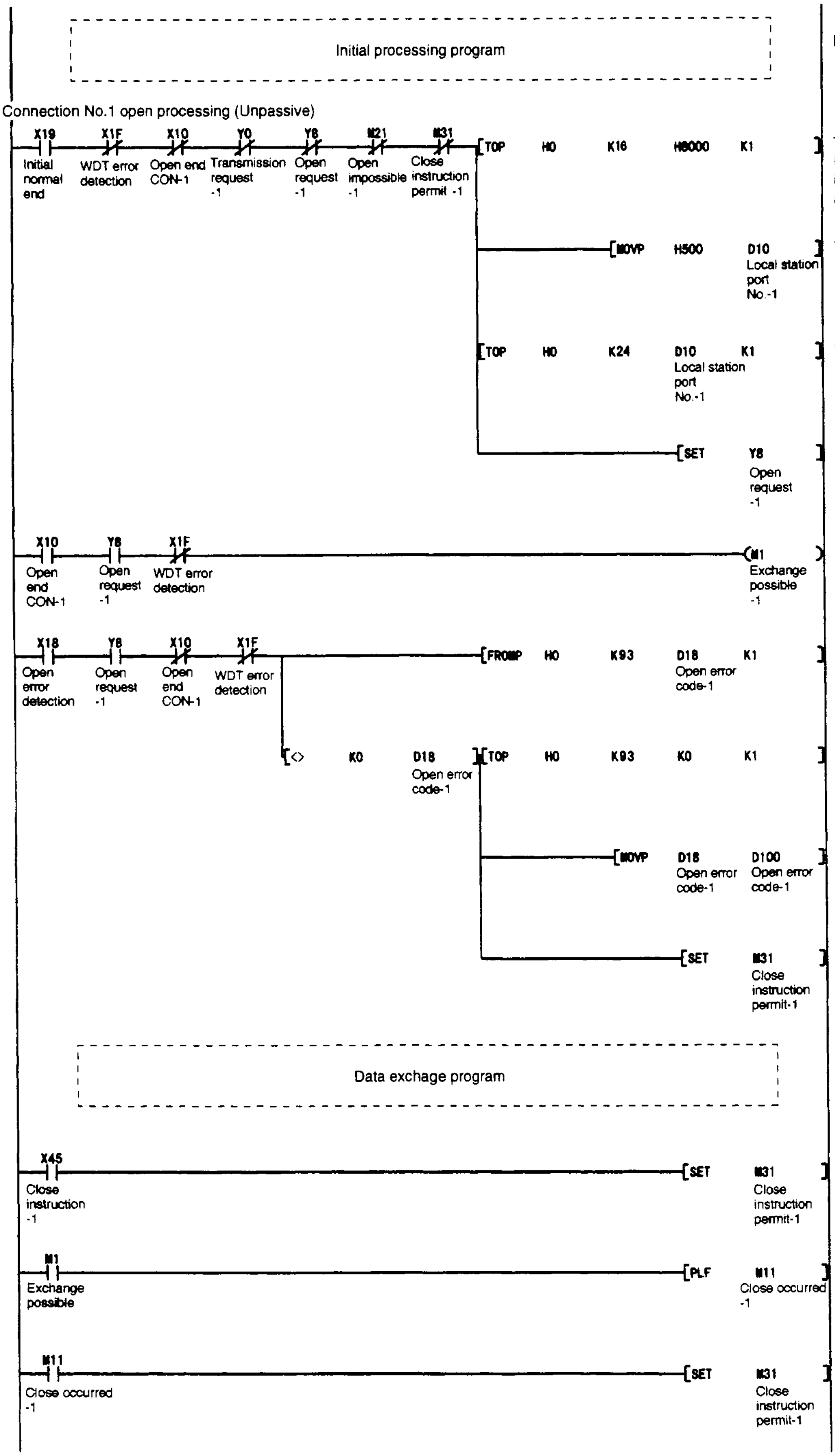
(1) The E71 is installed in the basic bases "0" slot.

(2) Exchange parameters are shown in the table below.

Exchange Parameter Name (When does not opening pairing)			Connection No.1	Connection No.2
Usage availability setting	Fixed buffer usage availability	Bit 0	0 : For transmission	1 : For reception
	Destination existence check	Bit 1	0 : Not performed	1 : Performed
	Pairing open	Bit 7	0 : Not performed	0 : Not performed
	Communication format	Bit 8	0 : TCP	0 : TCP
	Fixed buffer exchange procedure existence	Bit 9	0 : With procedure	0 : With procedure
	Open method	Bit 14 Bit 15	10 : Unpassive	10 : Unpassive
Exchange address setting	E71 port No.		500H	501H
	Remote node IP address		(Setting not required)	(Setting not required)
	Remote node port No.		(Setting not required)	(Setting not required)
	Remote node Ethernet address		(Default value)	(Default value)

Exchange Parameter Name (When opening pairing)			Connection No.1	Connection No.2
Usage availability setting	Fixed buffer usage availability	Bit 0	1 : For reception	0 : For transmission
	Destination existence check	Bit 1	1 : Performed	0 : Not performed
	Pairing open	Bit 7	1 : Performed	0 : Not performed
	Communication format	Bit 8	0 : TCP	0 : TCP
	Fixed buffer exchange procedure existence	Bit 9	0 : With procedure	0 : With procedure
	Open method	Bit 14 Bit 15	10 : Unpassive	10 : Unpassive
Exchange address setting	QE71 port No.		500H	(Setting not required)
	Remote node IP address		(Setting not required)	
	Remote node port No.		(Setting not required)	
	Remote node Ethernet address		(Default value)	

1 When not opening pairing



Refer to Items 5.3

The transmission, TCP, and Unpassive setting data is written in the connection No.1 usage availability area. (*1)

The E71 port No. is set to 500H.

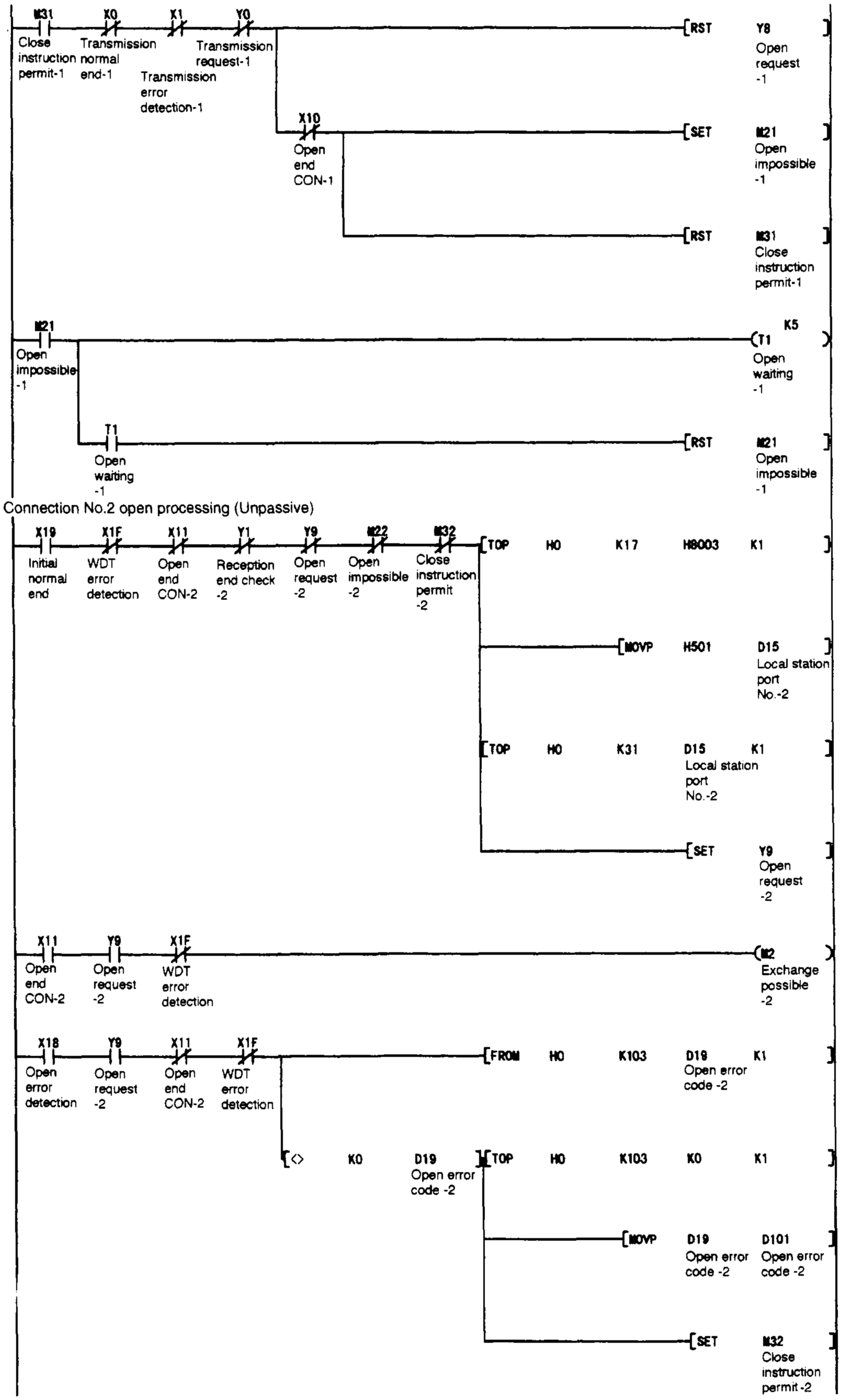
Written in the connection No.1 exchange address setting area.

The connection No.1 open error processing.

Refer to Chapters 6 to 8

Close request from local station occurred.

Close request from other node occurred.



Connection No.1 close processing.

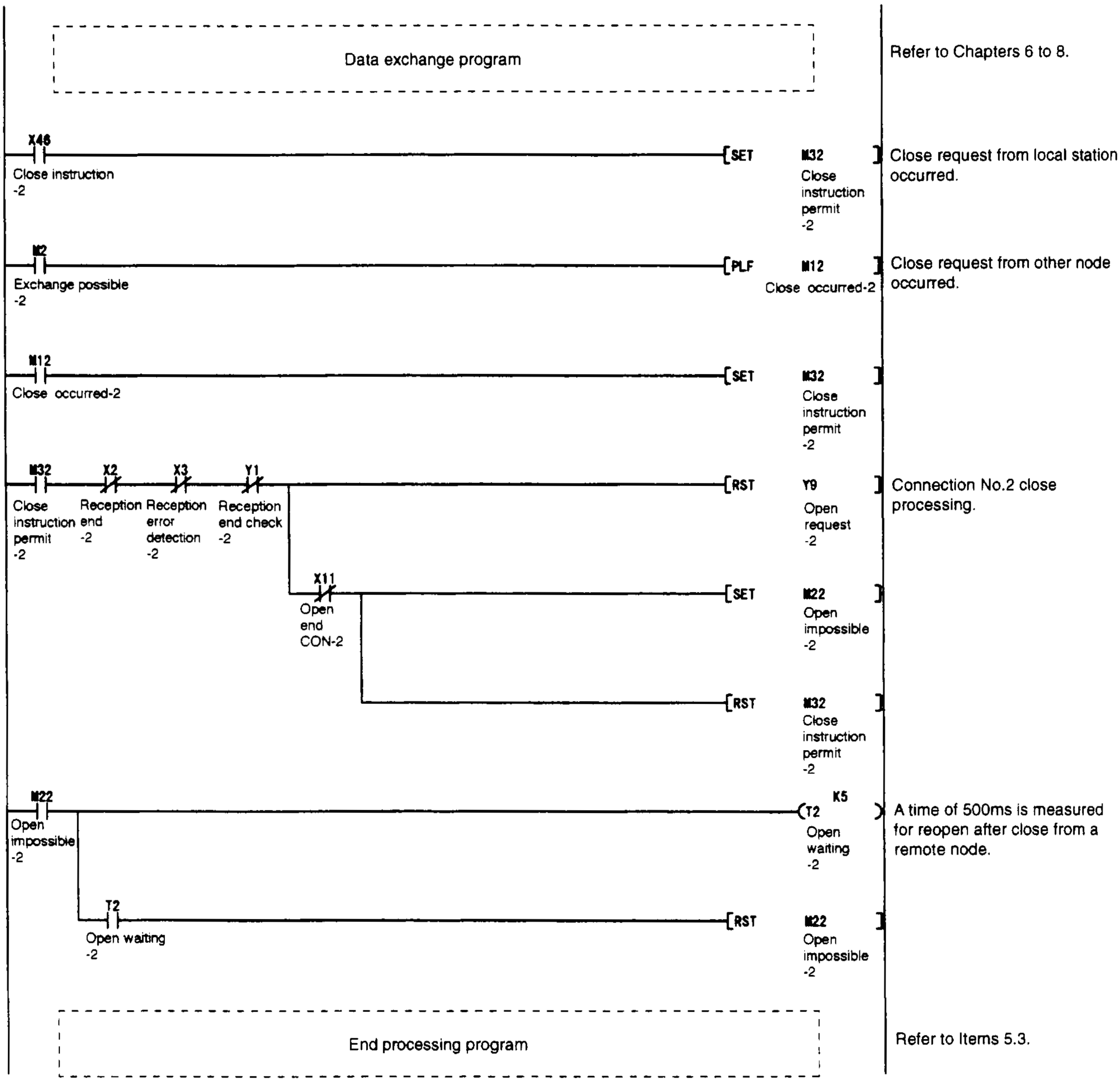
A time of 500ms is measured for reopen after close from a remote node.

The reception, TCP, and Unpassive setting data is written to the connection No.2 usage availability area settings area. (*1)

The E71 port No. is set to 501H.

Written to the connection No.2 exchange address settings area.

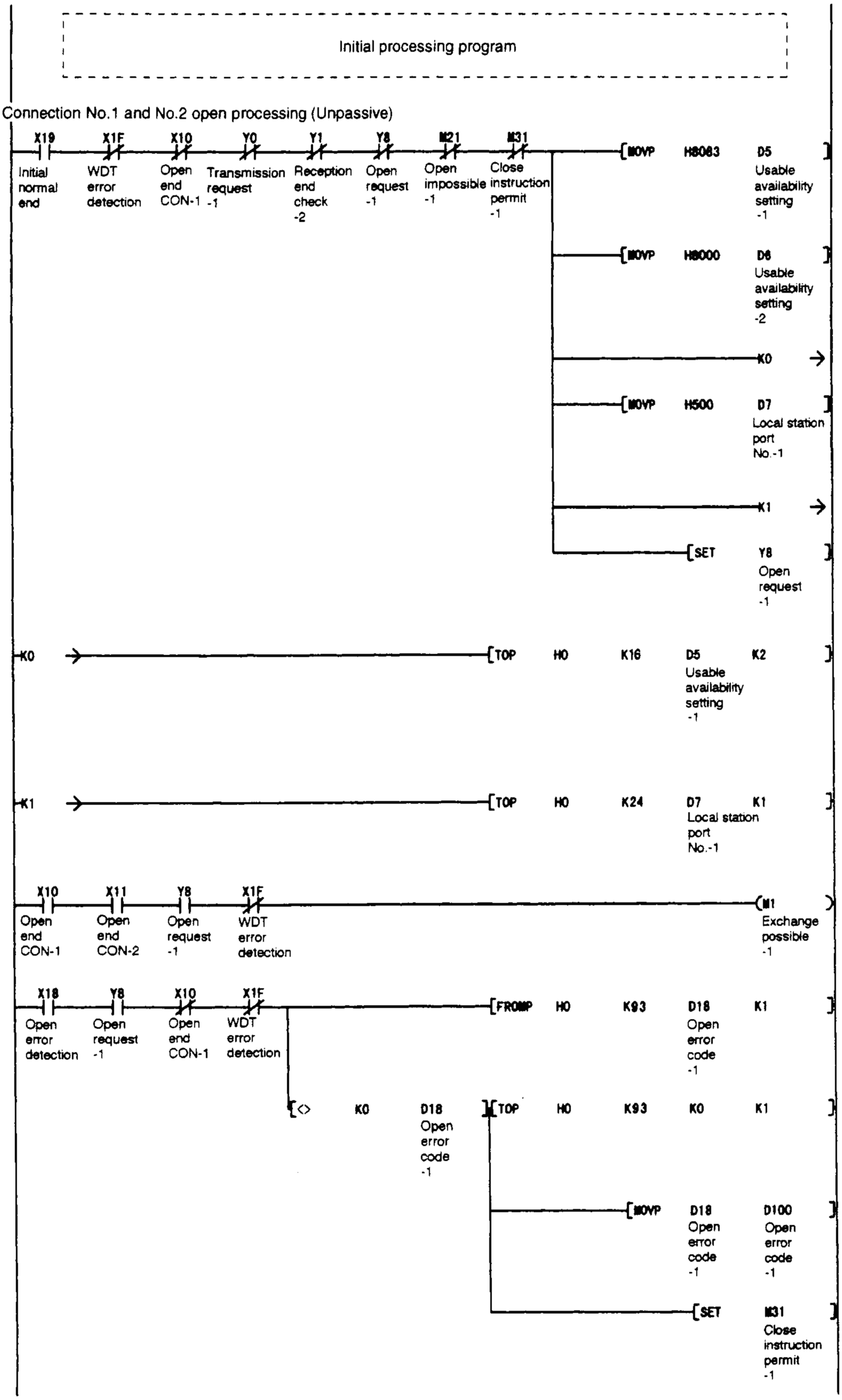
Connection No.2 open error processing.



*1 The data exchange shown below can be conducted after setting of usage availability with the value stored in D0 and D10.

	Fixed buffer exchange		Random access buffer exchange	Read/write data in the PLC CPU
	With procedure	Without procedure		
Connection No.1	Exchange enabled (Transmission only)	Exchange disabled	Exchange enabled	Exchange enabled
Connection No.2	Exchange enabled (Reception only)	Exchange disabled	Exchange enabled	Exchange enabled

2 When opening pairing



Refer to Items 5.3.

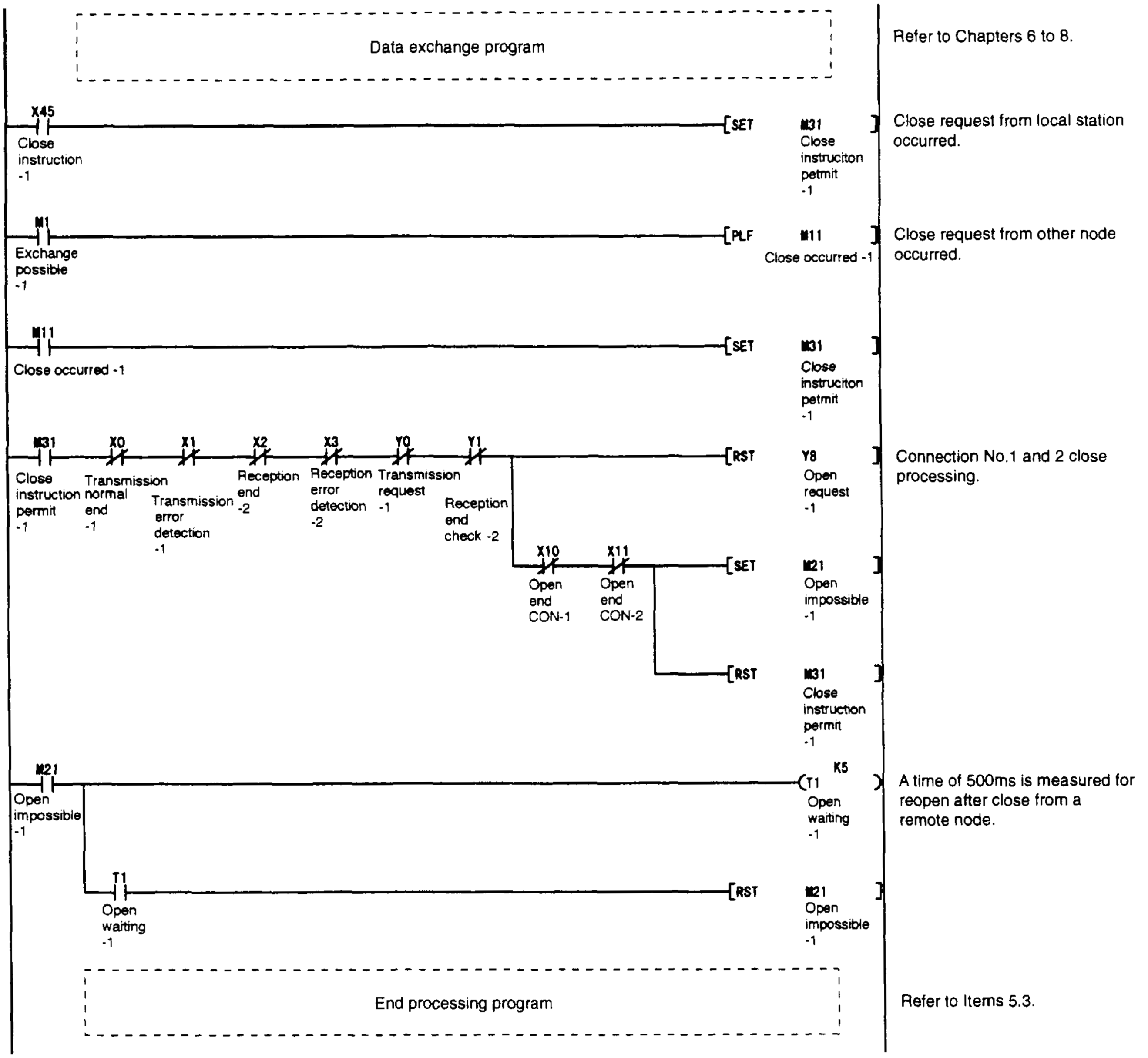
The reception, TCP, and Unpassive setting data is written in the connection No.1 usage availability area. (*2)

The transmission, TCP, and Unpassive setting data is written in the connection No.2 usage availability area. (*2)

The E71 port No. is set to 500H.

Written in the connection No.1 and No.2 exchange address setting area.

The connection No.1 open error processing.



*2 The data exchange shown below can be conducted after setting of usage availability with the value stored in D0 and D1.

	Fixed buffer exchange		Random access buffer exchange	Read/write data in the PLC CPU
	With procedure	Without procedure		
Connection No.1	Exchange enabled (Reception only)	Exchange disabled	Exchange enabled	Exchange enabled
Connection No.2	Exchange enabled (Transmission only)	Exchange disabled	Exchange enabled	Exchange enabled

5.5 Exchange State Storage Area

This section explains about the exchange state storage area where the initial processing state, open processing state, exchange state for each connection, and error log information are stored. Each processing result, exchange state for each connection, and error information can be checked by reading from this area.

5.5.1 Initial Processing State Storage Area

Buffer memory

(Address)	Initial Processing State Storage Area (6 words)		Default Value
50H (80)	Initial error code	(1 word)	0H (0)
51 to 52H (81 to 82)	Local station E71's IP address	(2 words)	0H (0)
53 to 55H (83 to 85)	Local station E71's Ethernet address	(3 words)	0H (0)

- After initial process end the corresponding values are stored in order.

1 **Initial error code (Default value = 0H) .... Address 50H (80)**

(a) Stores the error codes generated during initial processing execution.

(b) Please refer to Chapter 13 for details regarding initial processing error codes.

(c) Error codes are stored as binary values when the initial error detection signal (X1A) is on.

(d) The error codes are cleared when the initial normal end signal is on, but the following process can also be used to clear them.

① PLC CPU reset operation, or turning off the PLC power.

② Using the sequence program to write (0) in the initial error code storage error

2 **Local station E71's IP address (Default value = 0H)**
.. Address 51H to 52H....(81 to 82)

(a) Stores the E71's IP address set during the initial processing execution.

(b) The E71's IP address is stored as a binary value.

Example: The data storage condition when the IP address is A20009C0H (162.0.9.192) is shown below.

Address	Buffer memory
51H(81)	09C0H
52H(82)	A200H

3 **Local station E71's Ethernet address (Default value = 0H)**
.. Address 53H to 55H....(83 to 85)

(a) After initial processing, the E71's physical address is read from the ROM and stored. The Ethernet's physical address cannot be changed.

(b) The E71's Ethernet address is stored from the newest address in the L to H order.

5.5.2 Excange State Storage Area

Buffer memory

(Address)	Exchange State Storage Area (119 words)			Default Value
59H (89)	Local station E71's port No.		Information by Connection (10 words for connection No.1)	0H (0)
5A to 5BH (90 to 91)	Remote node IP address			0H (0)
5CH (92)	Remote node port No.			0H (0)
5DH (93)	Open error code			0H (0)
5EH (94)	Fixed buffer transmission/reception error code			0H (0)
5FH (95)	Fixed buffer exchange end code			0H (0)
60H (96)	Maximum value	Fixed buffer exchange's exchange time		0H (0)
61H (97)	Minimum value			0H (0)
62H (98)	Current value			0H (0)
63 to 6CH (99 to 108)	Local station E71's port No. to		Information by connection (For connection No.2)	(Same as above)
6D to 76H (109 to 118)	Local station E71's port No. to		Information by connection (For connection No.3)	(Same as above)
77 to 80H (119 to 128)	Local station E71's port No. to		Information by connection (For connection No.4)	(Same as above)
81 to 8AH (129 to 138)	Local station E71's port No. to		Information by connection (For connection No.5)	(Same as above)
8B to 94H (139 to 148)	Local station E71's port No. to		Information by connection (For connection No.6)	(Same as above)
95 to 9EH (149 to 158)	Local station E71's port No. to		Information by connection (For connection No.7)	(Same as above)
9F to A8H (159 to 168)	Local station E71's port No. to		Information by connection (For connection No.8)	(Same as above)

* After processes end from initial processing the corresponding values are stored in order.

1 **Local station E71's port No. (Default value = 0H) .... Address 59H....(89...)**

(a) Stores the port No. when the subject communication line was connected by open processing.

(b) The storage values are not set during the closed state.

2 **Remote node IP address (Default value = 0H) .... Address 5AH to 5BH....(90 to 91...)**

(a) Stores the partner remote node's IP address for when the subject communication line was connected using open processing.

(Example) The following is the data that is stored when the IP address is "A20009C0H."

Address	Buffer memory
5AH(90)	09C0H
5BH(91)	A200H

(b) The stored values are not set that are in the closed state.

3 **Remote node port No. (Default value = 0H) .... Address 5CH....(92...)**

(a) Stores the partner remote node port No. for the communication line connected by open processing.

(b) The stored value is not set in the closed state.

4 **Open error code (Default value = 0H) .... Address 5DH....(93...)**

(a) Stores the open processing results of the subject communication line.

(b) The open processing results are stored as binary values.

0 : Normal end

Other than 0 : Error end (refer to Chapter 13 for information regarding error control items.)

(c) Conducting the operation shown below will clear the error code.

① When reopening the connection for which the open error occurred. (When the open request signal is turned OFF to ON)

② When a PLC CPU reset operation was conducted or when the PLC power supply was turned off

5 **Fixed buffer transmission/reception error code (Default Value = 0H) .... Address 5EH....(94...)**

(a) Stores the error code (refer to Chapter 13 for details regarding error code items) when the error detection signal (X1, etc.) is turned on by the data transmission/reception with the remote node during fixed buffer exchange on the corresponding communication line.

(b) The transmission error codes are cleared under the following conditions.

- The fixed buffer transmission request/reception end check signal for the connection for which an error occurred is turned off.
- A reset operation is conducted for the PLC CPU or the PLC's power supply is turned off.

6 **Fixed buffer exchange end code (Default value = 0H) .... Address 5FH....(95...)**

(a) The error codes that are returned as a response from the remote node during fixed buffer exchange on the subject communication line are stored as binary values.

(b) The action to be taken by the end code in the response is conducted in accordance with the arrangements with the remote node.

(c) The end code can be cleared by conducting the following operations.

A reset operation is conducted for the PLC CPU or the PLC's power supply is turned off.

7 **Fixed buffer exchange's exchange time (Each default value = 0H) .... 60H to 62H....(96 to 98...)**

(a) Each of the fixed buffer exchange processing times (maximum value, minimum value, current value) are stored.

① Fixed buffer transmission processing time

The time from when the transmission request signal turns on to when the E71 conducts a transmission end.

(The processing time is not stored when the transmission error occurred.)

② Fixed buffer reception processing time

The time from when the reception end signal turns on to when the E71 ends in response to the response return processing from the remote node.

(b) The processing time is stored as a binary value in 10ms units.

(c) Each exchange time is changed to "0," when the subject communication line's open request signal (Y8 to YF) is changed from off to on.

5.5.3 Error Log Area

Buffer memory

(Address)	Error log area (11 words)		Default Value
A9H (169)	Error log	Area-1	0H (0)
AAH (170)		Area-2	0H (0)
ABH (171)		Area-3	0H (0)
ACH (172)		Area-4	0H (0)
ADH (173)		Area-5	0H (0)
AEH (174)		Area-6	0H (0)
AFH (175)		Area-7	0H (0)
B0H (176)		Area-8	0H (0)
B1H (177)		Area-9	0H (0)
B2H (178)		Area-10	0H (0)
B3H (179)		Area-11	0H (0)

(1) The following two areas are storage areas in which the codes that show the error contents are stored.

(Refer to Chapter 13)

① When the occurrence origin of the error that occurred cannot be checked

IP level error, reception data sum check error (TCP/UDP/IP check sum error).

② An error that occurred during random access buffer exchange or read/write of data in the PLC CPU.

Remarks

The error that occurred during the fixed buffer transmission is stored in the fixed buffer transmission error code (buffer memory addresses 94, 104, 114, ... 164) area.

(2) This error area has an 11 word data area and is configured of ring buffers that can store up to 10 units of error information. The next area to be set is normally set to 0000H and this makes it possible to determine what data is the newest.

(3) The value stored in this area is cleared when the power to the station installed in the E71 is turned on or when a reset operation is conducted. In addition, it can be cleared if the user writes a 0. It cannot be cleared using initial processing.

(4) It is not normally necessary to read this area, so read it when necessary when conducting maintenance.

5.5.4 Protocol Status Storage Areas

This is the area that is used to store the number of times of a protocol level that is used during exchange between the E71 and a remote node. (This is the count value controlled by the E71)

When the count value exceeds one word, the count is stopped at FFFFH (65535).

The storage values for all the protocol status storage areas can be cleared by the user writing a 0. In addition, they are cleared when the power is turned on to the station installed in the E71 and when a reset operation is conducted. (They are not cleared during initial processing.)

Normally it is not necessary to read this area, so read it when necessary during maintenance.

Buffer memory

(Address)	Exchange status storage area (80 words)		Default Value	
170H (368)	Number of times IP packets received	(1 word)	0H (0)	*
171H (369)	Number of times received IP packet discarded because of check sum error	(1 word)	0H (0)	*
172H (370)	Total number of transmitted IP packets	(1 word)	0H (0)	*
173 to 17FH (371 to 383)	System area (Use prohibited)	(13 words)	———	
180H (384)	Total number of received ICMP	(1 word)	0H (0)	*
181H (385)	Number of times received ICMP packet discarded because of check sum error	(1 word)	0H (0)	*
182H (386)	Total number of transmitted ICMP packets	(1 word)	0H (0)	*
183H (387)	Total number of received ICMP echo request packets	(1 word)	0H (0)	*
184H (388)	Total number of transmitted ICMP echo reply packets	(1 word)	0H (0)	*
185H (389)	Total number of transmitted ICMP echo request packets	(1 word)	0H (0)	*
186H (390)	Total number of received ICMP echo reply packets	(1 word)	0H (0)	*
187 to 18FH (391 to 399)	System area (Use prohibited)	(9 words)	———	
190H (400)	Total number of received TCP packets	(1 word)	0H (0)	*
191H (401)	Number of times received TCP packet discarded because of check sum error	(1 word)	0H (0)	*
192H (402)	Total number of transmitted TCP packets	(1 word)	0H (0)	*
193 to 19FH (403 to 415)	System area (Use prohibited)	(13 words)	———	
1A0H (416)	Total number of received UDP packets	(1 word)	0H (0)	*
1A1H (417)	Number of times received UDP packet discarded because of check sum error	(1 word)	0H (0)	*
1A2H (418)	Total number of transmitted UDP packets	(1 word)	0H (0)	*
1A3 to 1BFH (419 to 447)	System area (Use prohibited)	(29 words)	———	

*This shows the areas that can be cleared with a 0 from the sequence program.

5.6 Data Exchange during the PLC CPU is Stopped

This section explains the settings etc., that make it possible to continue data exchange from remote nodes to the E71 even after the PLC CPU in the station installed in the E71 enters the STOP status and the E71 open request signals (Y8 to YF) are turned off.

Data exchange while the PLC CPU is stopped can be conducted using the settings shown in Item 5.6.1 after the initial processing and open processing from the PLC CPU are completed.

> **Point**
>
> Be sure to match the system specifications when conducting data exchange while the PLC CPU is stopped.

5.6.1 Settings for Continuing Data Exchange

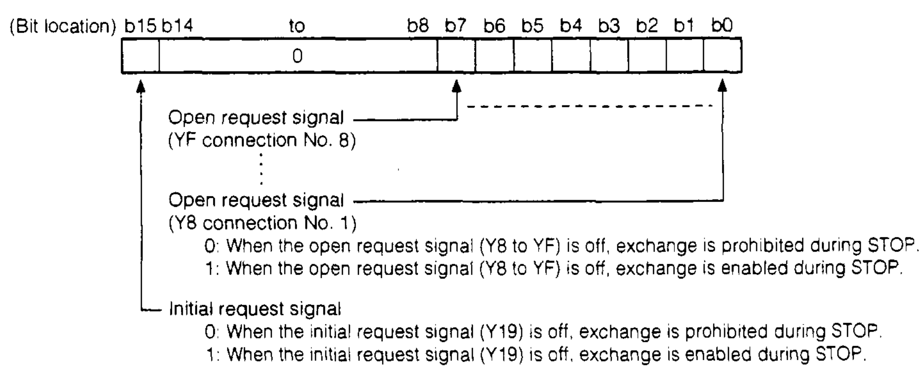
This section explains the settings for using the data exchange function while the PLC CPU is stopped.

The relationship between the data exchange when the PLC CPU is stopped and this setting is shown in Item 5.6.3.

1 Setting method

The setting that uses the data exchange function while the PLC CPU is stopped is done using the buffer memory exchange specification during STOP area (address 1F0H).

2 Setting value for the exchange specification during STOP area



3 Specification of the setting value to the exchange instruction area during STOP

To continue exchange after the initial request signal (Y19) and open request signals (Y8 to YF) are turned off because the PLC CPU is stopped, etc., turn on the bit 15 of the exchange instruction area during STOP.

(Example) To continue exchange between connections No.1 and No.2 after the initial request signal (Y19) and open request signal (Y8, Y9) are turned off, set "8003H".

> **Point**
>
> (1) When continuing the exchange with remote nodes while the PLC CPU is at the stop state, always set the bit 15 of the above exchange instruction area during STOP to "1 (on)" .
>
> (2) When continuing the exchange when the initial request signal (Y19) is on and the open request signal (Y8 to YF) is turned from on to off, turn on the bit for corresponding connection No. of the exchange instruction area during STOP.
>
> (Example) To continue the exchange between connections No. 1 and No. 2 when the initial request signal (Y19) is on and the open request signals (Y8, Y9) are off, set "0003H".

5.6.2 Functions for Which Continuing Data Exchange is Possible

When conducting the setting to exchange data while the PLC CPU is stopped, the following shows the functions that make it possible to continue data exchange between a remote node and the E71 even after the PLC CPU of the station installed in the E71 enters the STOP status and the E71 open request (Y8 to YF) and initial request signal (Y19) have turned off.

Data exchange function	Data exchange while the PLC CPU is stopped
Fixed buffer exchange (with procedure, without procedure)	Not possible
Random access buffer exchange	Possible
Exchange for reading/writing data in the PLC CPU	Possible

5.6.3 Relationship between the Setting and Data Exchange during the PLC CPU is Stopped

The setting that performs the data exchange while the PLC CPU is stopped after the initial processing and open processing from the PLC CPU is completed is valid.

After the setting for data exchange while the PLC CPU is stopped is valid, the PLC CPU enters the stop status, and the E71 initial request signal (Y19) and open request signals (Y8 to YF) are turned off, it is possible to continue data exchange from the remote node to the E71 while the PLC CPU is stopped.

The setting used for the data exchange function while the PLC CPU is stopped, the I/O signal (initial request signal and open request signal) with the PLC CPU, and the relationship for data exchange while the PLC CPU is stopped are shown below.

A concrete example is shown on the next page.

1 Data exchange is possible while the PLC CPU is stopped when exchange enable during STOP is set using the buffer memory's exchange specification during STOP area (address 1F0H).

2 When exchange enable during STOP is set, the change from on to off of the initial request signal (Y19) and the open request signals (Y8 to YF) are ignored.

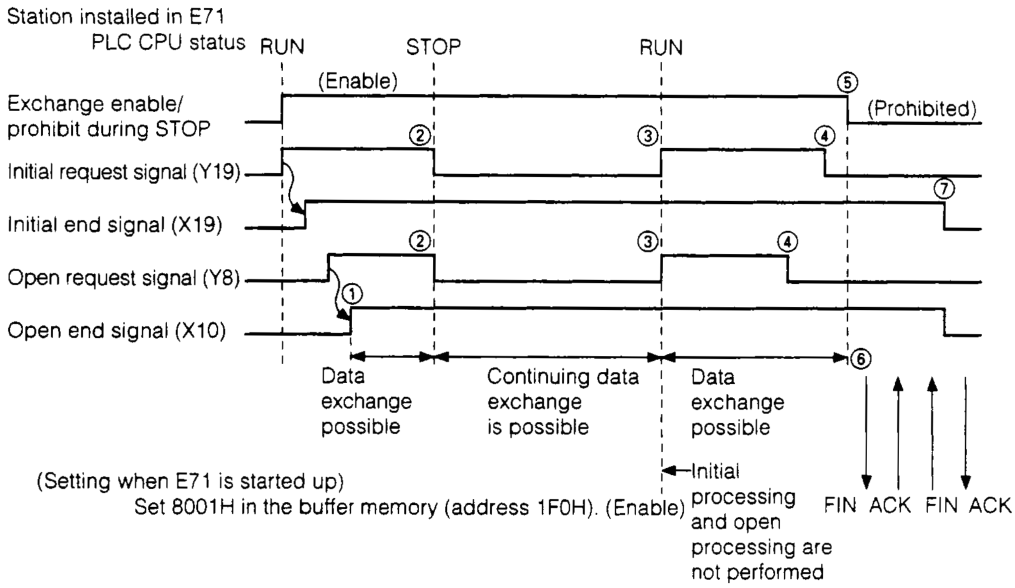
	QE71 processing	
	OFF → ON	ON → OFF
Initial request signal (Y19)	Initial processing is conducted. (*1)	End processing is not conducted.
Open request signal (Y8 to YF)	Open processing is conducted. (*1)	Close processing is not conducted.

*1 Only when the signal is turned on first.

Point

When the buffer memory's exchange specification during STOP area setting is the default value, then data exchange cannot be conducted when the PLC CPU is stopped. Conduct data exchange by conducting initial processing, open processing, close processing, and end processing in accordance with the procedure shown in Items 5.1 to 5.4.

1 When the communication circuit is opened by a setting to allow data exchange through connection No. 1 while the PLC CPU is stopped



① Data exchange with a remote node is possible by turning on the open end signal (X10). (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

② The PLC CPU of the station installed in the E71 enters the STOP status and the initial signal (Y19) and open request signal (Y8) turn off. Close processing and end processing are not performed because exchange enable during STOP is set. Data exchange can be continued. (Random access buffer exchange and exchange of read/write data in the PLC CPU are possible.)

③ The PLC CPU of the station installed in the E71 enters the RUN status and the initial request signal (Y19) and open request signal (Y8) turn on. Initial processing and open processing are not conducted because exchange enable during STOP is set. Data exchange can be continued. (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

④ The open request signal (Y8) and initial request signal (Y19) are turned off. Close processing and end processing are not conducted because exchange enable during STOP is set. Data exchange can be continued. (Random access buffer exchange and exchange of read/write data in the PLC CPU are possible.)

⑤ Change the setting to exchange prohibited during STOP. Set the buffer memory (address 1F0H) to 0000H.

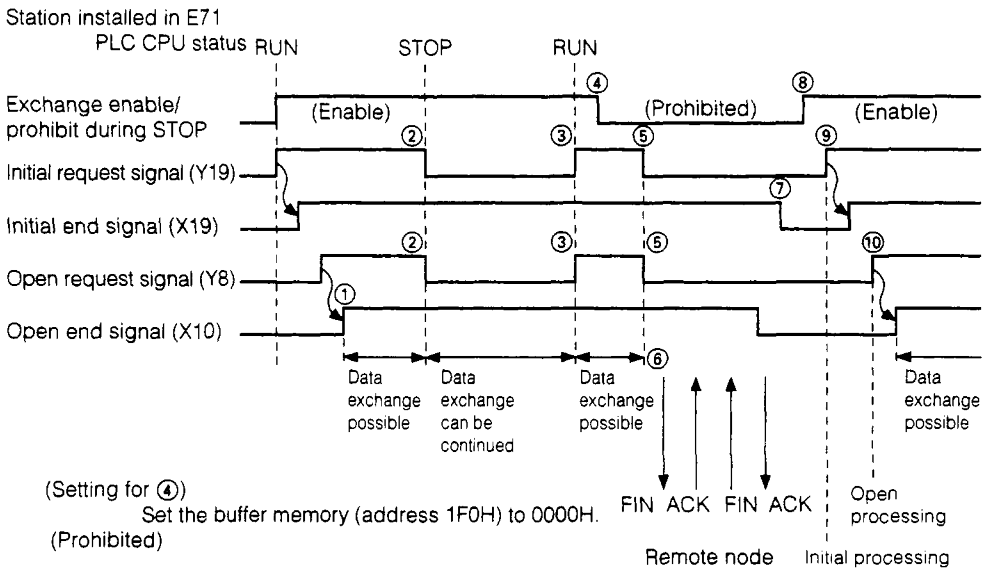
⑥ Exchange circuit close processing is conducted after the open request signal (Y8) is turned off after the setting has been changed to exchange prohibit during STOP.

⑦ End processing is conducted after the initial request signal (Y19) is turned off after the setting is changed to exchange prohibit during STOP.

2 When initial processing and open processing are reperformed when the setting for data exchange while the PLC CPU is stopped is changed midway for connection No. 1

(Changed from exchange data while the PLC CPU is stopped to do not exchange data while the PLC CPU is stopped.)

(a) When the open request signal (Y8) and the initial request signal (Y19) are turned off after the setting is changed to exchange prohibited



① Data exchange to a remote node becomes possible by turning the open end signal (X10) on. (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

② The PLC CPU in the station installed in the E71 enters the STOP status and the initial request signal (Y19) and the open request signal (Y8) turn off. Close processing and end processing are not conducted because exchange enable during STOP is set. Data exchange can be continued. (Random access buffer exchange and exchange of read/write data in the PLC CPU are possible.)

③ The PLC CPU in the station installed in the E71 enters the RUN state and the initial request signal (Y19) and open request signal (Y8) are turned on. Initial processing and open processing are not conducted because the exchange enable during STOP is set. Data exchange can be continued. (Fixed buffer exchange, random access exchange, and exchange of read/write data in the PLC CPU are possible.)

④ Change the setting to exchange prohibited during STOP. Set the buffer memory (address 1F0H) to 0000H. Close processing and end processing are not conducted because the initial request signal (Y19) and open request signal (Y8) are turned on. Data exchange can be continued. (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

⑤ The initial request signal (Y19) and open request signal (Y8) are turned off.

⑥ Close processing is conducted because the exchange prohibited during STOP is set.

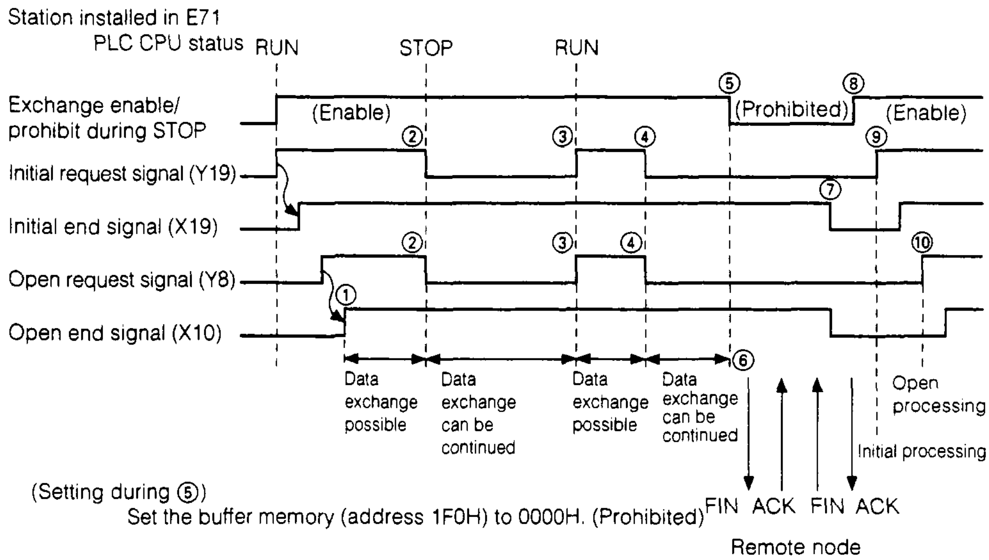
⑦ End processing is conducted for the same reason as in ⑥ above.

⑧ Change the setting to exchange enable during STOP. Set the buffer memory (address 1F0h) to 8001H. Initial processing and open processing are not conducted because the initial request signal (Y19) and open request signal (Y8) are turned off.

⑨ The initial request signal (Y19) is turned on to reconduct initial processing. E71 initial processing is conducted.

⑩ The request signal (Y8) is turned on to reconduct open processing. Open processing of the exchange circuit with the remote node is conducted. (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

(b) When changing the setting to exchange prohibited after turning the open request signal (Y8) and initial request signal (Y19) off



① Data exchange with the remote node can be conducted by turning the open end signal (X10) on. (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

② The PLC CPU of the station installed in the E71 enters the STOP status, and the initial request signal (Y19) and open request signal (Y8) are turned off. Close processing and end processing are not conducted because exchange enable during STOP is set. Data exchange can be continued. (Random access buffer exchange and exchange of read/write data in the PLC CPU are possible.)

③ The PLC CPU in the station installed in the E71 enters the RUN status and the initial request signal (Y19) and open request signal (Y8) are turned on. Initial processing and open processing are not conducted because the exchange enable during STOP is set. Data exchange can be continued. (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

④ The initial request signal (Y19) and open request signal (Y8) are turned off. Close processing and end processing are not conducted because exchange enable during STOP is set. Data exchange can be continued. (Random access buffer exchange and exchange of read/write data in the PLC CPU are possible.)

⑤ Change the setting to exchange prohibited during STOP. Set the buffer memory (address 1F0H) to 0000H.

⑥ Communication line close processing is conducted after the setting is changed to exchange prohibit during STOP because the open request signal (Y8) is turned off.

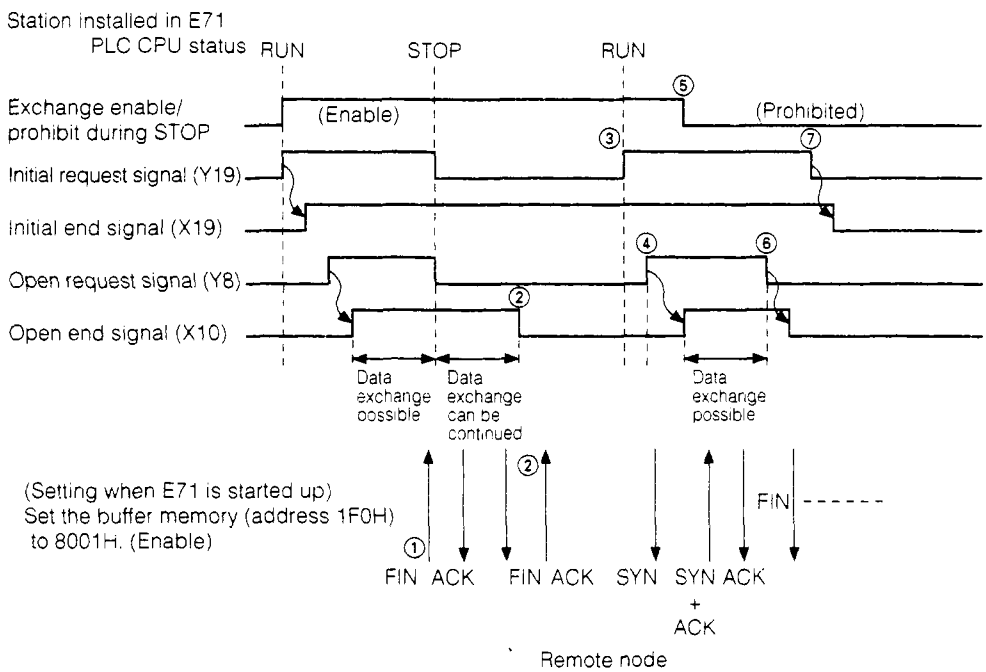
⑦ End processing is conducted because the initial request signal (Y19) is turned off after the setting is changed to exchange prohibited during stop.

⑧ Change the setting to exchange enable during STOP. Set the buffer memory (address 1F0H) to 8001H.

⑨ The initial request signal (Y19) is turned on to reconduct initial processing. E71 initial processing is conducted.

⑩ The open request signal (Y8) is turned on to reconduct open processing. Open processing of the communication line with the remote node is conducted. (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

3 **When a close request is received from the partner remote node when data is being exchange because the setting allows data exchange through connection No. 1 while the PLC CPU is stopped.**



① A close request is received from the partner remote node when data is exchanged with the remote node using the data exchange function when the PLC CPU is stopped.

② Close processing is conducted in the open end signal (X10) is turned off. Data exchange cannot be conducted.

* FIN is transmitted even if the open request signal (Y8) is turned off.

③ The PLC CPU of the station installed in E71 enters the RUN status and the initial request signal (Y19) is turned on. Initial processing is not conducted because the exchange enable during STOP is set.

④ Open request signal (Y8) turns on and open processing is conducted and then the open end signal (X10) is turned on. Data exchange with the remote node becomes possible. (Fixed buffer exchange, random access buffer exchange, and exchange of read/write data in the PLC CPU are possible.)

⑤ Change the setting to exchange prohibited during STOP. Set the buffer memory (1F0H) to 0000H. Close processing and end processing are not conducted because the initial request signal (Y19) and open request signal (Y8) are turned on. Data exchange can be continued.

⑥ Communication circuit close processing is conducted because the open request signal (Y8) turns off after the setting is changed to exchange prohibited during STOP.

⑦ End processing is conducted when the initial request signal (Y19) turns off after the setting is changed to exchange prohibited during STOP.

FIXED BUFFER EXCHANGE SECTION

The fixed buffer exchange section explains the with procedure data exchange methods and the without procedure data exchange method when data is exchanged with remote node external equipment that uses the Ethernet interface module's fixed buffer and with the PLC CPU.

Fixed buffer exchange is begun after initial processing and open processing that is described in Chapter 5 connects the communication line.

In addition, conduct close processing and end processing during data exchange end on the subject communication line.

When conducting fixed buffer exchange with procedures, read Chapter 6.
When conducting fixed buffer exchange without procedures, read Chapter 7.

6. FIXED BUFFER EXCHANGE WITH PROCEDURE

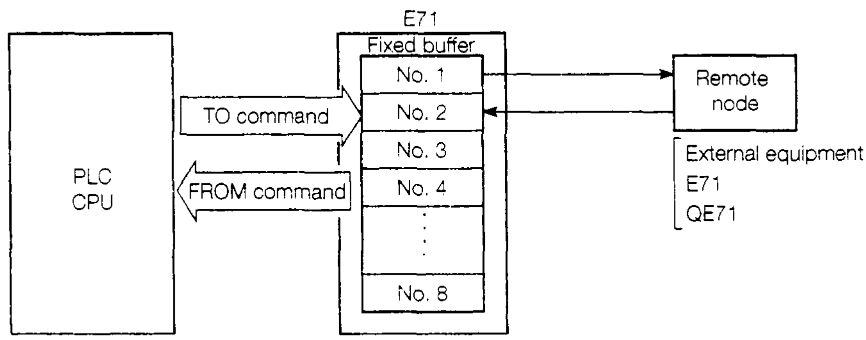
This section explains the method used to conduct exchange with a remote node with procedures using the E71's fixed buffer.

6.1 Control Format

This section explains the control format used for fixed buffer exchange with procedure.

Remote node exchange processing using the fixed buffer is conducted during the handshake with the remote node for data transmission from the PLC CPU and data reception from the remote node.

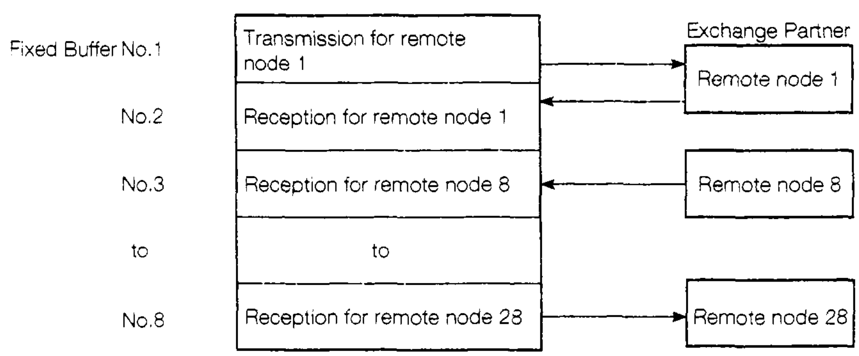
(1) The exchange processing data flow is as follows.



(2) Data exchange can be conducted with remote nodes in the Ethernet to which E71 is connected and with remote nodes that are connected by a router relay function (Refer to Chapter 12). As is shown in the diagram below, the various fixed buffers (No. 1 to No. 8) are used to set the remote node with which to exchange and the usage availability (for transmission and for reception, with procedures and without procedures, etc.) using the E71's communication line open (Refer to Item 5.5) to set the exchange partner for each buffer.

① When TCP/IP is used, a fixed buffer exchange partner setting using the parameter settings becomes valid when the E71's open end signal changes from OFF to ON during boot-up. The exchange partner cannot be changed while the open end signal is on.

② When UDP/IP is used, the fixed buffer exchange partner can be changed after open processing. (It is possible to change the exchange parameter's remote node IP address and remote node port No., but is not possible to change the local station's E71's port No.)



Point

(1) When with procedures is selected during opening, random access buffer exchange, and reading and writing data to the PLC CPU exchange can be conducted at the same time as fixed buffer exchange (transmission or reception) with procedure for the subject connection. (Refer to Item 5.1 (*1))

(2) When changing the exchange partner, do not conduct pairing setting (Refer to Item 5.4.1 1 (b) ③) and existence check setting (Refer to Item 5.3.1). If these settings are made the E71 will not operate correctly.

(3) The transmission and reception processing during data transmission and reception is given below.

① During transmission

When the transmission request signal (Y0 to Y7) is ON, the E71 transfers the subject fixed buffer data to the remote node set in the subject area with a buffer memory address of 18H to 4FH (24 to 79). (*1)

② During reception

If there is reception from the remote node set in the subject area for the buffer memory addresses 18H to 4FH (24 to 79), the E71 will conduct reception processing. (*1)

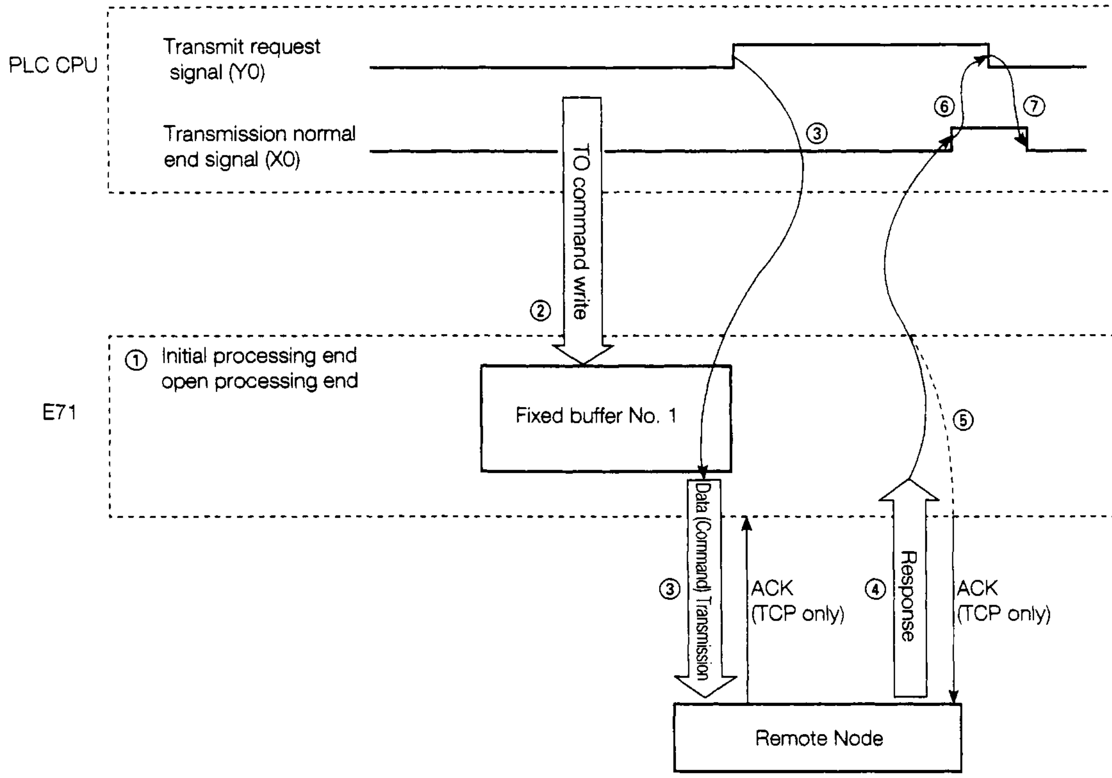
In addition, when the reception data is stored in the subject fixed buffer during reception processing, the E71 updates the subject connection's remote node IP address and remote node port No. for the buffer memory addresses 59H to A8H (89 to 168).

If there is a reception from a remote node that is not set in the buffer memory addresses 18H to 4FH (24 to 79), the E71 will ignore the reception data.

*1 During TCP/IP unpassive open, data is transmitted to and received from the remote nodes stored in the subject area for the buffer memory addresses 59H to A8H (89 to 168).

6.1.1 Transmission Control Method

This section explains about the control method when data is transmitted to the remote node by the E71 using an example where the fixed buffer No. 1's data is transmitted to a remote node.



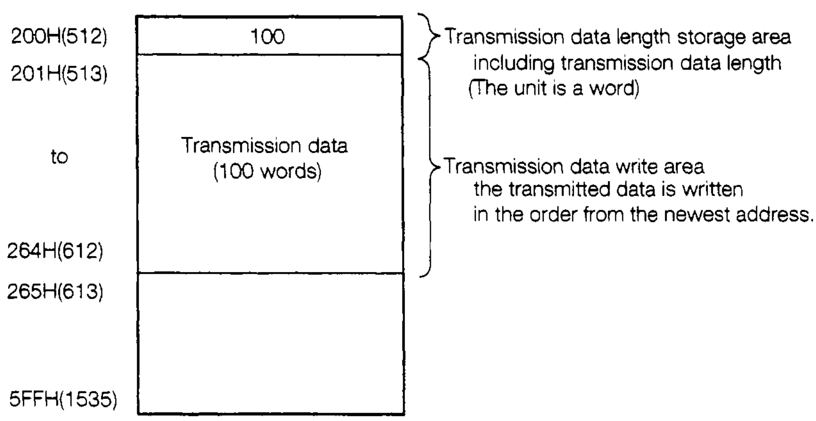
① The E71 initial processing is conducted. (Refer to Items 5.2 to 5.3)

Line open processing with a remote node is conducted. (Refer to Item 5.4)

② The sequence program's TO command writes the transmission data length and transmission data in the E71's fixed buffer.

The transmission data length is written to the subject fixed buffer's first address (512). The transmission data is written to the subject fixed buffer's first address + 1.

The following diagram shows an example of a 100-word transmission using fixed buffer No. 1.



③ Changing the transmission request signal (Y0) to ON using the sequence program transmits the data to the node (from the parameter settings) that is specified by the fixed buffer (No.1).

④ When data is received from the E71 by the specified remote node, a response is returned to the E71.

⑤ The E71 turns ON the transmission normal end signal (X0) when the response is received from the remote node.

⑥ When the transmission normal end signal turns on, the sequence program turns OFF the transmission request signal (Y0).

⑦ The transmission normal end signal is automatically turned OFF when the transmission request signal is turned to OFF.

Point

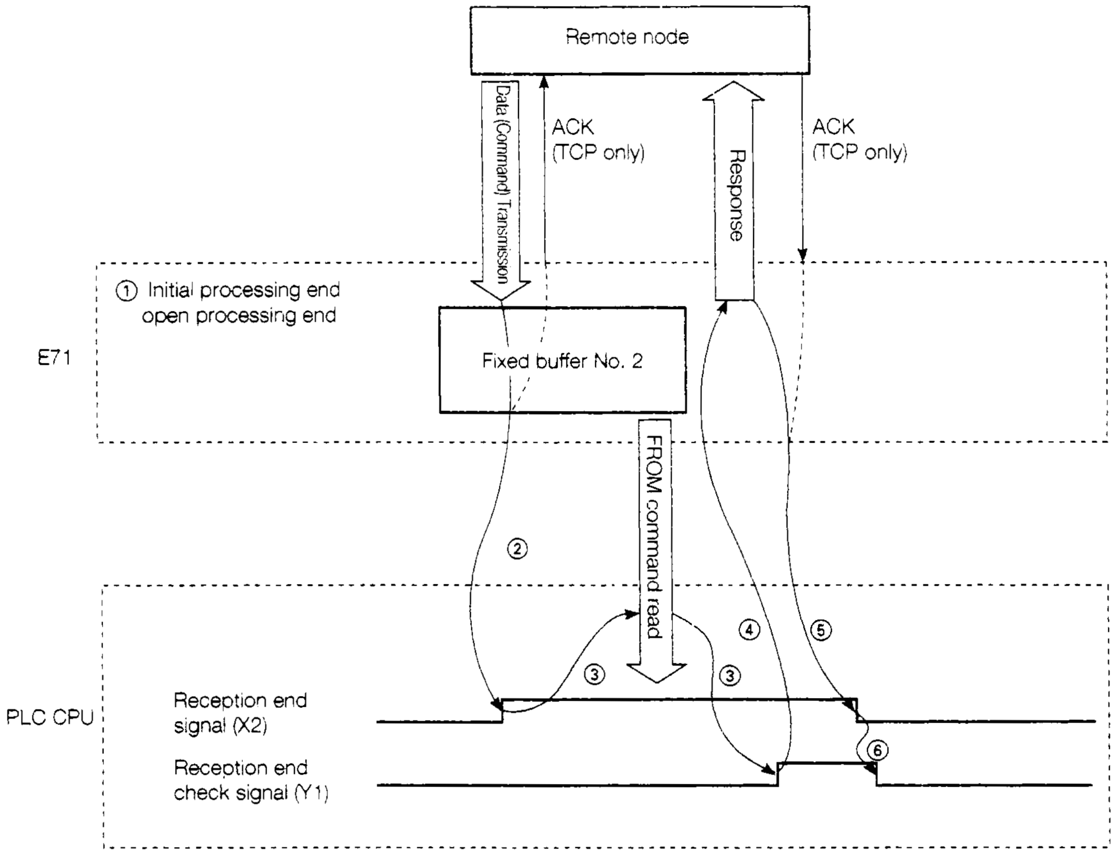
(1) At the following times the transmission error detection signal (X1) turns ON so conduct retransmission processing when the transmission request signal turns from OFF to ON after the transmission error processing has ended.
   ① When a response is not received within the response monitoring timer value.
   ② When the response end code is anything other than "00H."
(2) This shows the processing when the open request signal and initial request signal are turned off during transmission when the E71 "Function for Data Exchange during the PLC CPU is Stopped" (Item 5.6) is not used.
   ① The E71 conducts close processing after the transmission processing end when the open request signal (Y8) turns OFF during transmission.
   ② The E71 conducts close processing and end processing after transmission processing end when the initial request signal (Y19) turns OFF during transmission.

Remarks

For more details on cases when the reception of Close/Abort (RST) has occurred from the partner remote node prior to the completion of the data communication performed immediately before, refer to Remarks in 2 of Item 5.4.3.

6.1.2 Reception Control Method

This section explains the control method that the E71 receives the data from the remote node receiving data from the remote node to the fixed buffer No.2 as an example.



① The E71 initial processing is conducted. (Refer to Items 5.2 to 5.3)
The remote node and line opening processing. (Refer to Item 5.4)
To conduct fixed buffer exchange, initial processing and open processing must be completed.

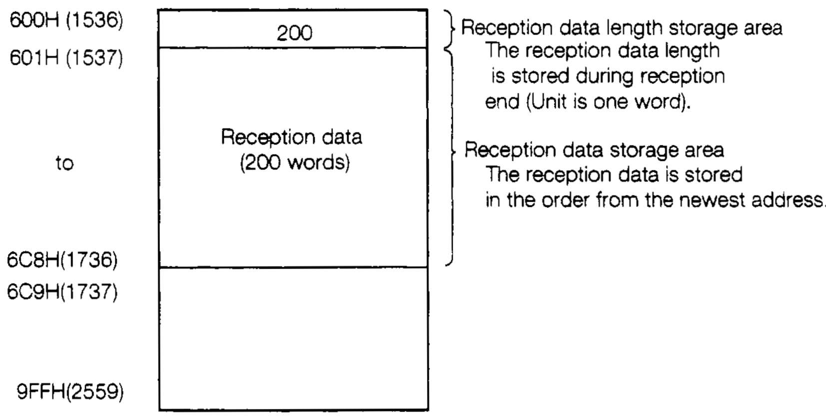
② When the data received from the remote node by parameter settings is stored in the fixed buffer (No.2), the QE71 turns the reception end signal (X2) ON.

The reception data length and reception data are stored in the fixed buffer.

The reception data length is stored in the subject fixed buffer's first address (1536).

The reception data is stored in the subject fixed buffer's first address + 1 order.

The following diagram shows an example of a 200-word reception using fixed buffer No. 2.



③ The reception data length and reception data stored in the fixed buffer are read by the sequence program's FROM command when the reception end signal is turned ON.

At the same time the reception end check signal (Y1) is turned ON by the sequence program.

④ The E71 returns a response to the remote node (by the parameter settings) when the reception end check signal is turned ON.

⑤ When the response returned is ended, the E71 automatically turns OFF the reception end signal.

⑥ The reception end check signal is turned OFF by the sequence program when the reception end signal is turned OFF.

Point

(1) The reception end signal (X2) does not turn ON during error data reception.
In addition, the data is not stored in fixed buffer No. 2.

(2) This shows the processing when the open request signal and initial request signal are turned off during reception when the E71 "Function for Data Exchange during the PLC CPU is Stopped" (Item 5.6) is not used.

① When the open request signal (Y9) turns OFF during reception, the E71 immediately performs close processing.

② When the initial request signal (Y19) turns OFF during reception, the E71 immediately conducts close processing and end processing.

Remarks

For more details on cases when the reception of Close/Abort (RST) has occurred from the partner remote node prior to the completion of the data communication performed immediately before, refer to Remarks in 2 of Item 5.4.3.

6.2 Data Format

This shows the data item order and contents for exchange data (commands and responses) between the E71 and a remote node when conducting fixed buffer exchange with procedures.

As shown below, the exchange data consists of a header and application data.

Header	Application data
	Maximum 2040 bytes

As shown below, the data code of the application data can be expressed as either binary or ASCII code.

The DIP switch on the front of the E71 is used to set either binary or ASCII. (For details regarding the setting method refer to Item 4.3.2.)

6.2.1 Format When Exchanging with Binary Code

The command response data item order for when conducting fixed buffer exchange with procedures when exchanging the application data portion of the exchange data as binary code data are shown below.

1 Transmission/reception data order when exchanging using TCP/IP

① Order when transmitting/receiving commands

Header			Application data		
Ethernet	IP	TCP	Subheader	Data length setting (L) (H)	Text (command)
			60_H 00_H		
(14 bytes)	(20 bytes)	(20 bytes)	(2 bytes)	(2 bytes)	(Maximum 1017 words)

② Order when transmitting/receiving responses

Header			Application data	
Ethernet	IP	TCP	Subheader	End code
			$E0_H$	
(14 bytes)	(20 bytes)	(20 bytes)	(1 byte)	(1 byte)

2 Transmission/reception data order when exchanging using UDP/IP

① Order when transmitting/receiving commands

Header			Application data		
Ethernet	IP	UDP	Subheader	Data length setting (L) (H)	Text (command)
			60_H 00_H		
(14 bytes)	(20 bytes)	(8 bytes)	(2 bytes)	(2 bytes)	(Maximum 1017 words)

② Order when transmitting/receiving responses

Header			Application data	
Ethernet	IP	UDP	Subheader	End code
			$E0_H$	
(14 bytes)	(20 bytes)	(8 bytes)	(1 byte)	(1 byte)

6.2.2 Format When Exchanging with ASCII Code

The command and response data item order when conducting fixed buffer exchange with procedure when exchanging the application data portion of exchange data using ASCII code data is shown below.

1 **Transmission/reception data order when exchanging using TCP/IP**

① Order when transmitting/receiving commands

Header			Application data		
Ethernet	IP	TCP	Subheader	Data length setting	Text (command)
			"6" "0" "0" "0"	(H) (L)	
			36_H 30_H 30_H 30_H		
(14 bytes)	(20 bytes)	(20 bytes)	(4 bytes)	(4 bytes)	(Maximum 1016 words)

② Order when transmitting/receiving responses

Header			Application data	
Ethernet	IP	TCP	Subheader	End code
			"E" "0"	(H) (L)
			45_H 30_H	
(14 bytes)	(20 bytes)	(20 bytes)	(2 bytes)	(2 bytes)

2 **Transmission/reception data order when exchanging using UDP/IP**

① Order when transmitting/receiving commands

Header			Application data		
Ethernet	IP	UDP	Subheader	Data length setting	Text (command)
			"6" "0" "0" "0"	(H) (L)	
			36_H 30_H 30_H 30_H		
(14 bytes)	(20 bytes)	(8 bytes)	(4 bytes)	(4 bytes)	(Maximum 1016 words)

② Order when transmitting/receiving responses

Header			Application data	
Ethernet	IP	UDP	Subheader	End code
			"E" "0"	(H) (L)
			45_H 30_H	
(14 bytes)	(20 bytes)	(8 bytes)	(2 bytes)	(2 bytes)

6.2.3 Exchange Data Item Contents

The command and response data item contents when fixed buffer exchange with procedure is conducted is shown below.

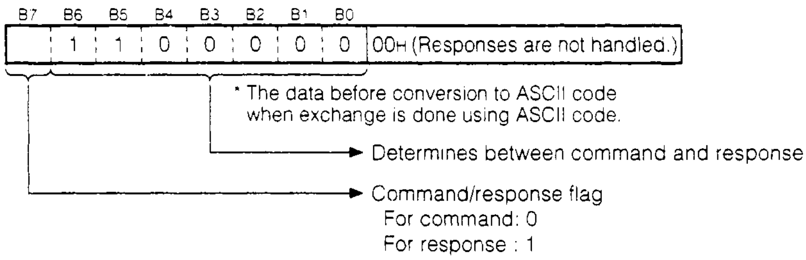
1 Header

The header is the header that is used for TCP/IP and UDP/IP. For the E71, since the E71 can be added or removed the user does not need to make the setting.

2 Subheader

The subheader format is as shown below.

For E71, since the E71 can be added or removed the user does not need to make the setting.



The subheader data code order when conducting fixed buffer exchange is as follows.

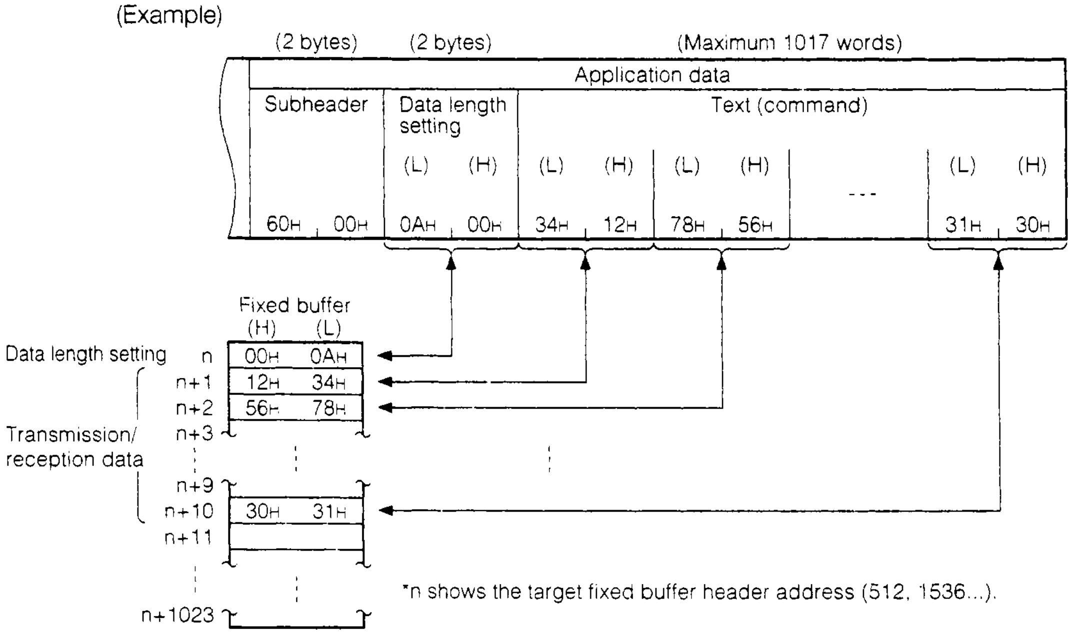
	Subheader code during exchange	
	For command exchange	For response exchange
When exchanging binary code	60H 00H	E0H
When exchanging using ASCII code	36H 30H 30H 30H	45H 30H

3 Data length setting and text (command)

The data length setting shows the text data capacity in number of words.

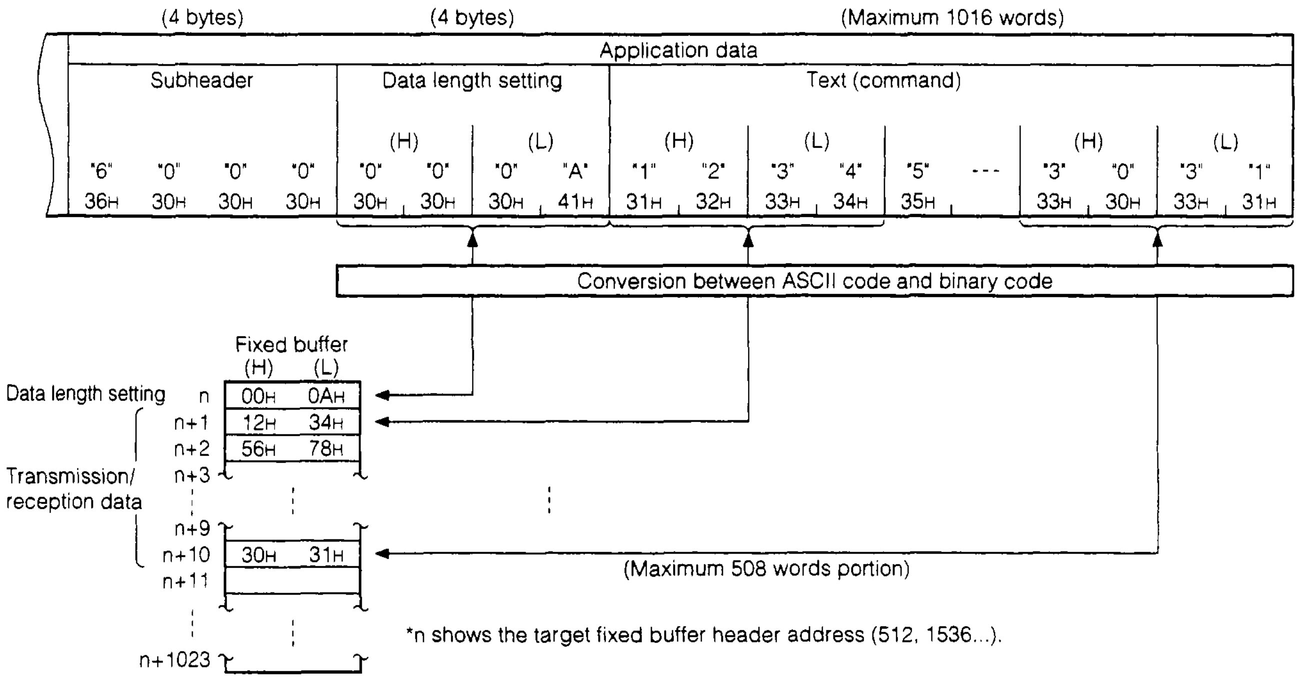
The text (command) shows the data for the data length setting portion that is sent to the exchange partner node.

① Data length setting and text (command) portion format when exchanging with binary code (Example)



② Data length setting and text (command) portion format when exchanging with ASCII code

(Example)



Point

(1) The maximum exchange data capacity that can be handled by the PLC CPU when binary code is specified is 1017 words. The data length setting range is 1 to 1017. The unit is a word.

(2) The maximum exchange data capacity that can be handled by the PLC CPU when ASCII code is specified is 508 words. This is an exchange data amount that is approximately one-half that of when binary code is specified. The data length is communicated using ASCII code ("0001" to "01FC") when the number of words is expressed in hexadecimal notation. The setting range is 1 to 508. The unit is a word.

4 End code

This shows the end code that is added to the response when conducting fixed buffer exchange. The end code is stored in the buffer memory's exchange status storage area.

When binary code is specified		When ASCII code is specified	
00H	Normal end	30H30H	Normal end
50H	Command/response classification undefined error	35H30H	Command/response classification undefined error
52H	Number of data words error	35H32H	Number of data word error
——		35H34H	ASCII conversion error

For details regarding error codes refer to Chapter 13.

6.3 Programming

This section explains programming method for using the fixed buffer to conduct exchange between the E71 and a remote node with procedures.

6.3.1 Programming Creation Precautions

(1) Fixed buffer exchange can only be conducted when the open request signal (Y8 to YF) and the open end of signal (X10 to X17) is turned ON. Initial processing and communication line open processing must be completed. (Refer to Chapter 5)

(2) The parameter settings are entered into the E71 when the open request signal (Y8 to YF) turns from OFF to ON during boot up. Except for those cases shown in the following diagram (3), the control contents cannot be changed even if the parameter contents are written over while the open end signal (X10 to X17) is ON.

(3) When using a connection opened by UDP, the exchange parameters setting area's exchange address setting area setting values can be changed before data is transmitted or received, and the exchange partner remote node can be switched. Therefore, data can be transmitted in order to multiple remote nodes, so to prevent exchange trouble from occurring, switch the partner remote node and conduct transmission and reception.

(4) The data link that is specified (stored) in the buffer memory when exchange with procedures is conducted, uses units of one word. If the buffer memory transmission data link exceeds the range during transmission, there will be an exchange error and the transmission will not be conducted.

(5) When data is received from a fixed buffer, be sure the reception end check signal (Y0 to Y7) is ON during reception end (the point at which the reception end signal turns ON).

> A response is returned to the remote node when the reception end check signal turns ON, and the following reception data is stored in the subject fixed buffer. If the reception end check signal does not come ON, a response is not returned to the remote node, so an exchange error occurs at the remote node end.

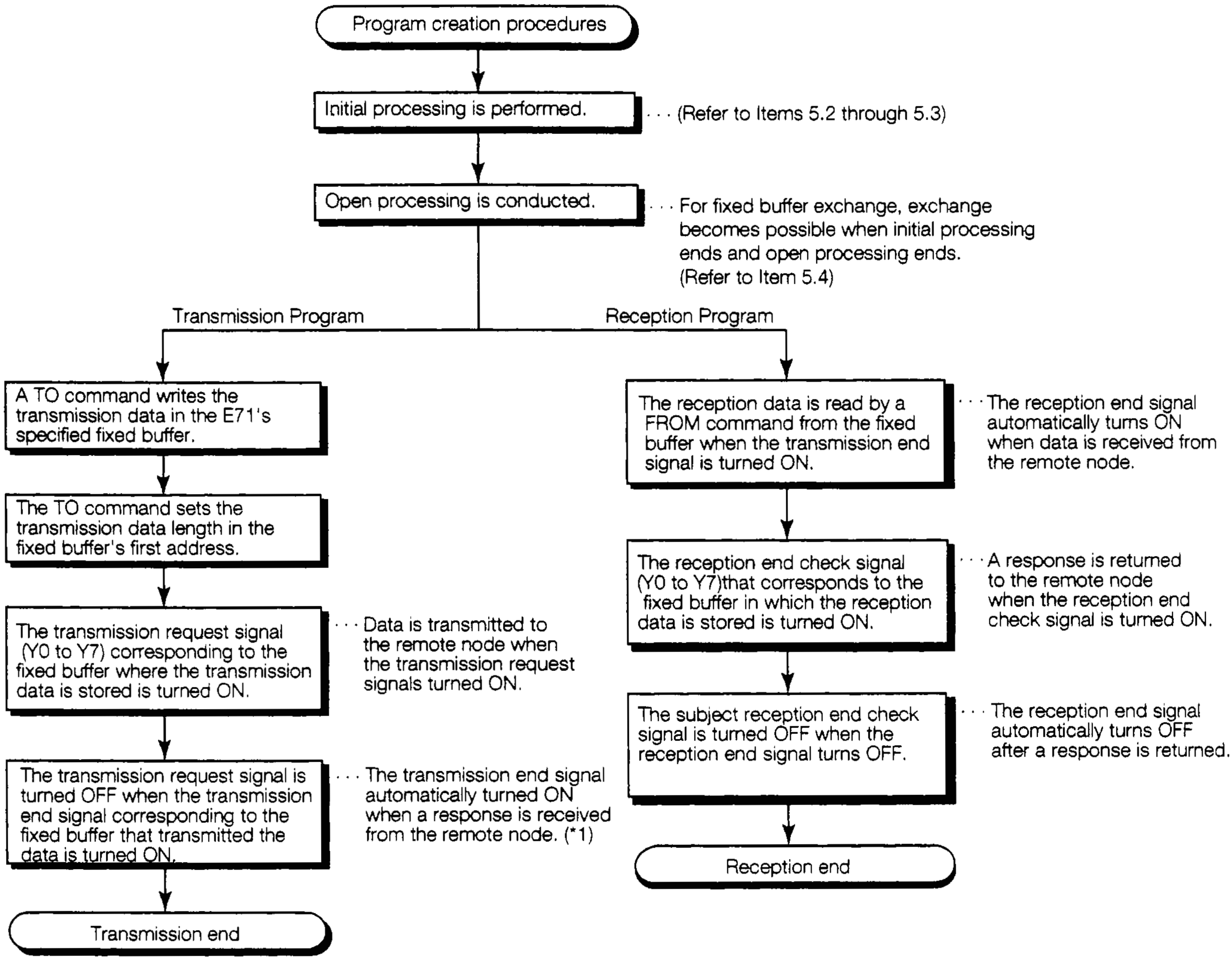
(6) When receiving data from remote nodes the data length in the command being transmitted from the remote node must not exceed the range shown in Item 6.2.3 $\boxed{3}$ Point.

When the E71 receives a command with a data length that exceeds the settable range, it might conduct closed processing for the corresponding connection without transmitting a response. Check using the I/O signal's open end signal (X10 to X17) or the open error code area (address 5DH, 67H, ..., error code 71H) in the buffer memory's exchange status storage area.

(7) For data (command) transmission, the next data (command) should be sent after the completion of data communication (such as after the reception of a response) for the transmission of the previous data (command).

6.3.2 Program Creation Procedure

This section explains the fixed buffer data transmission and reception program creation procedures.



*1 When the transmission error detection signal turns ON, the information for individual connections in the transmission state storage area (transmission error code, end code) is handled as described in Chapter 13.

6.3.3 Example Fixed Buffer Exchange Program (With Procedure)

This section explains the programming method for conducting data exchange with a remote node using a fixed buffer.

(Program Conditions)

(1) All connection exchange parameters are set to those parameters specified in Item 5.4.5.

(2) The fixed buffer transmission data is set to D300 to D399.

(3) The fixed buffer reception data is stored in D500 to D599.

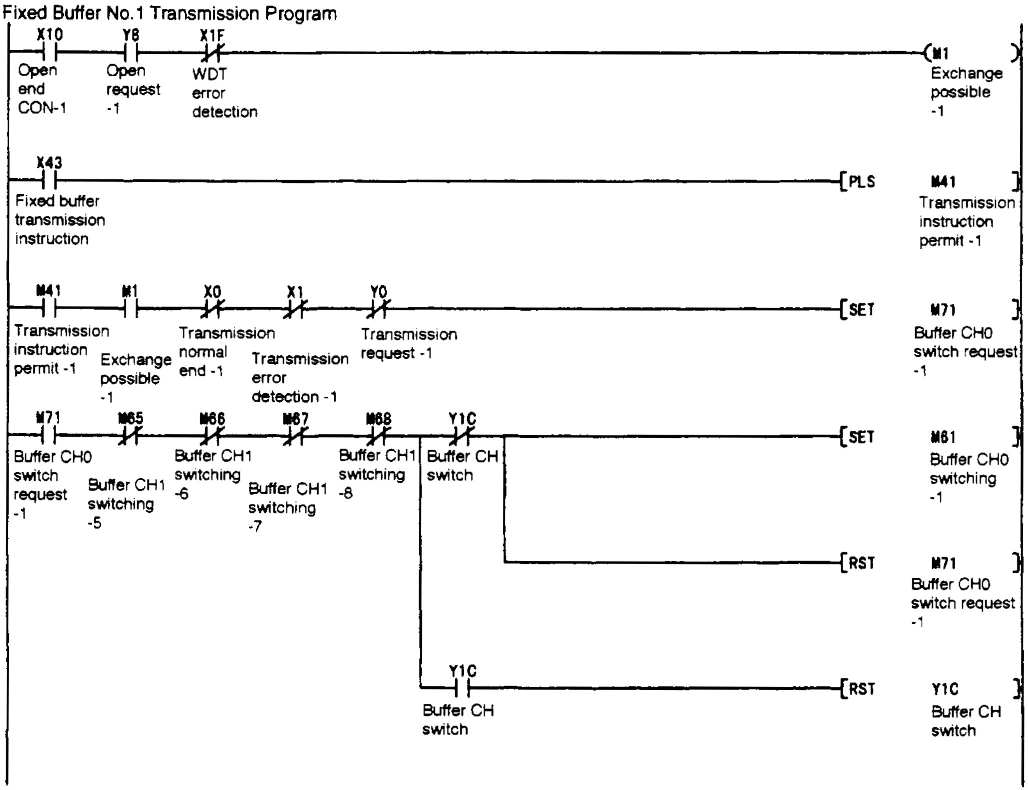
(4) The error code and end code storage destination is allotted as follows.

D108 Transmission error code

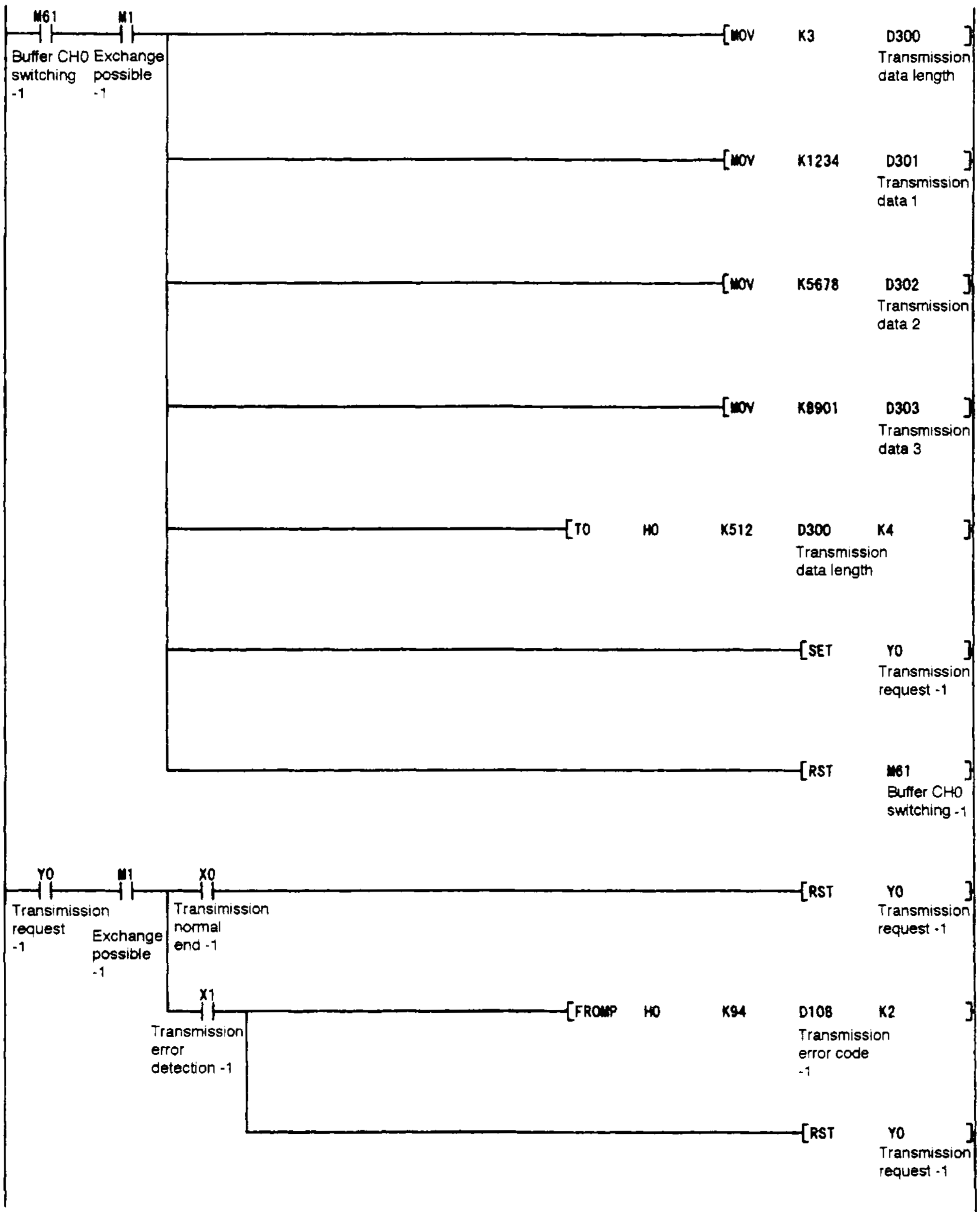
D109 Reception end code

D110 Reception error code

Fixed Buffer No.1 Transmission Program



Channel switch for buffer memory completed.
Writing to the fixed buffer of connection 1 is possible.



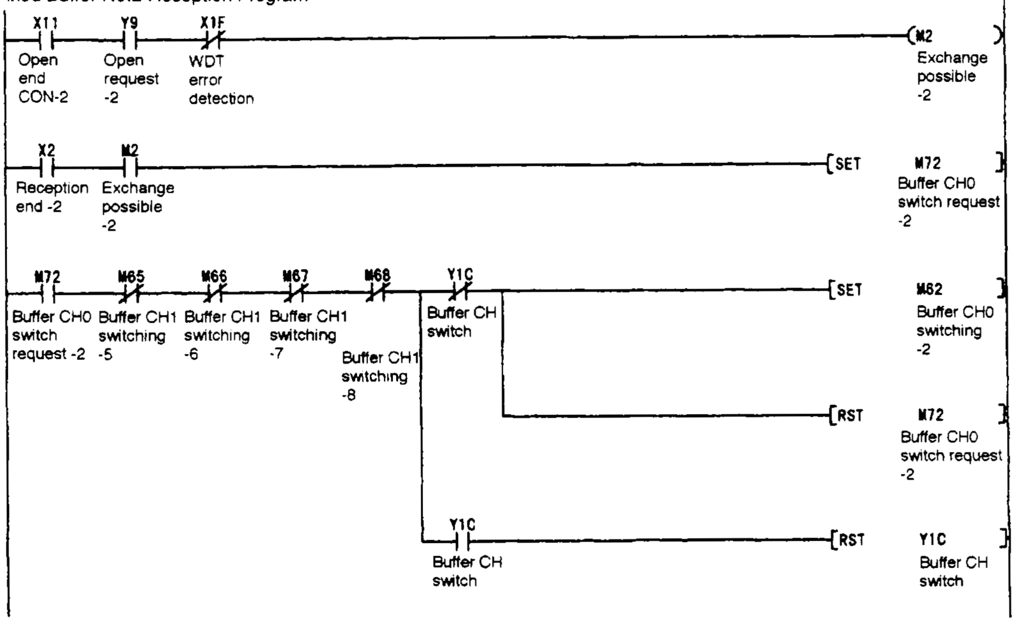
Set the data length to 3 words.

Set the transmission data.

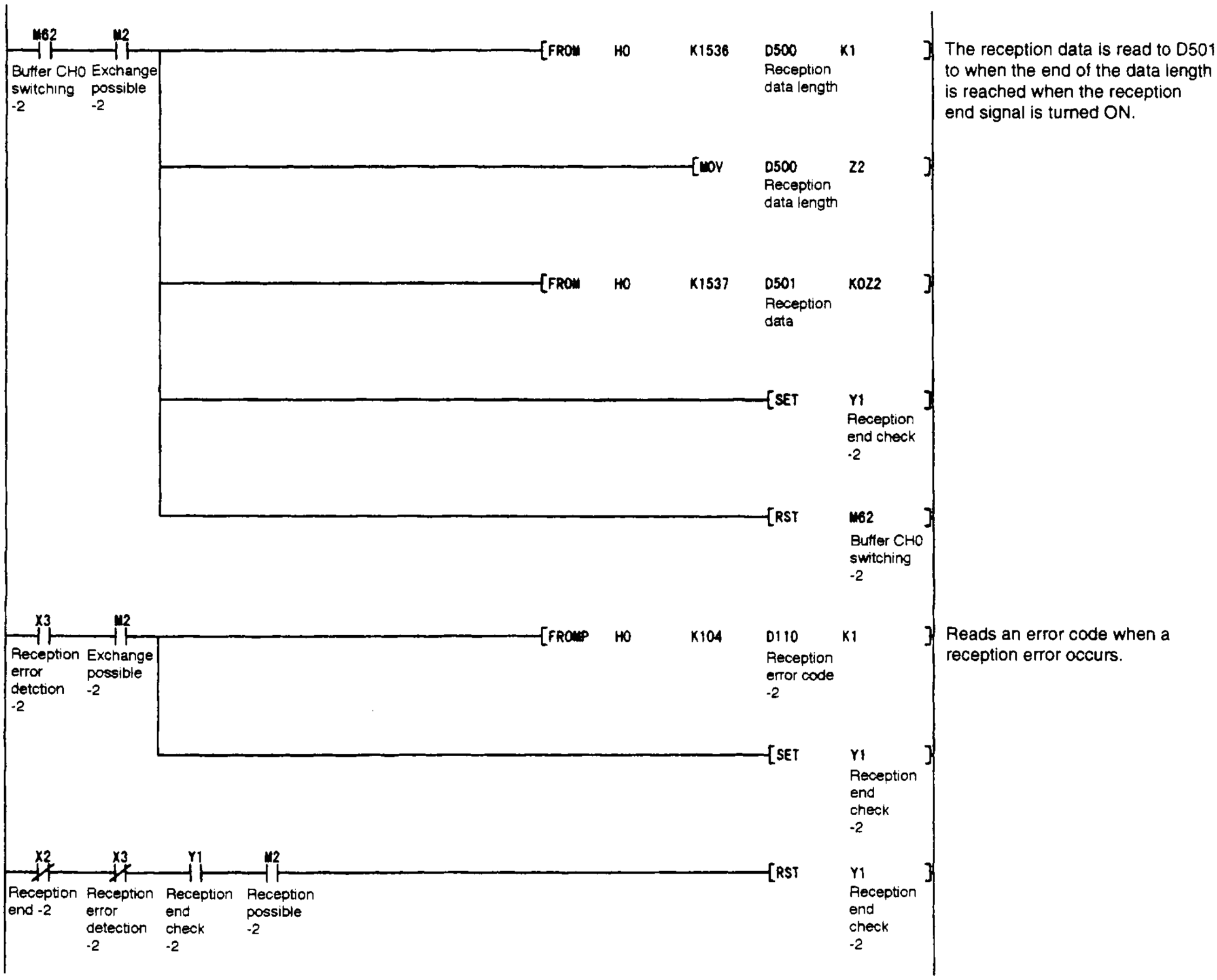
Writes the data length and transmission data in the fixed buffer No.1's area.

Reads error infromation when a transmission error occurs.

Fixed Buffer No.2 Reception Program



Channel switch for buffer memory completed.
Reading from the fixed buffer of connection 2 is possible.



* When the I/O control method of the PLC CPU of the station installed in the E71 is the refresh method and a fixed buffer read/write is performed after the buffer memory channel switching signal (Y001C) is switched from on → off/off → on, a read/write must be performed after the channel switching signal (Y001C) is output to the E71. Output the channel switching signal (Y001C) to the E71 using the tail's *1 of Item 3.6.2.

MEMO

7. FIXED BUFFER EXCHANGE WITHOUT PROCEDURE

This section explains the method for exchanging with a remote node without procedure using the E71's fixed buffer.

Point

Following is an overview of the difference from fixed buffer exchange using with procedure.

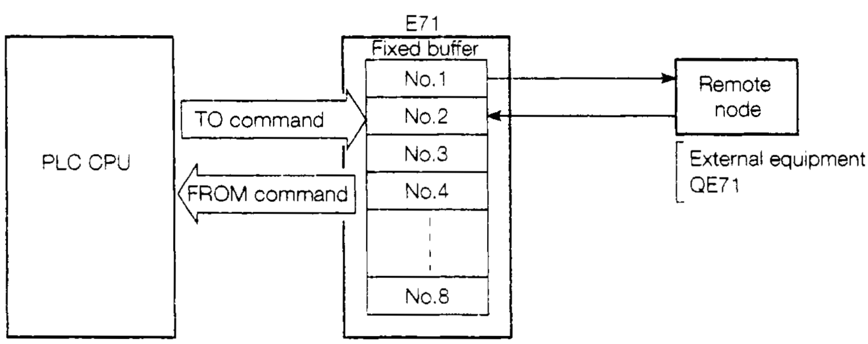
① During data transmission, the data is transmitted to the fixed buffer without the message application adding a subheader or a data length. During data reception, the header is removed from the received message and all the data is stored in the fixed buffer.
② A response to the data reception is not transmitted.
③ Conducts exchange in binary code regardless of the DIP switch SW2 setting on the front of the E71.
④ The application data portion that can be handled by one exchange is a maximum of 2046 bytes.
⑤ The subject connection becomes for fixed buffer exchange without procedure special use. Fixed buffer exchange with procedure, random access buffer exchange, and reading and writing data to the PLC CPU exchange are not conducted at the same time as fixed buffer exchange without procedure.

7.1 Control Format

This section explains the control format used to conduct fixed buffer exchange without procedure.

Fixed buffer remote node exchange processing can be conducted without procedure for data transmission from the PLC CPU and data reception from a remote node.

(1) The exchange processing data flow is shown below.



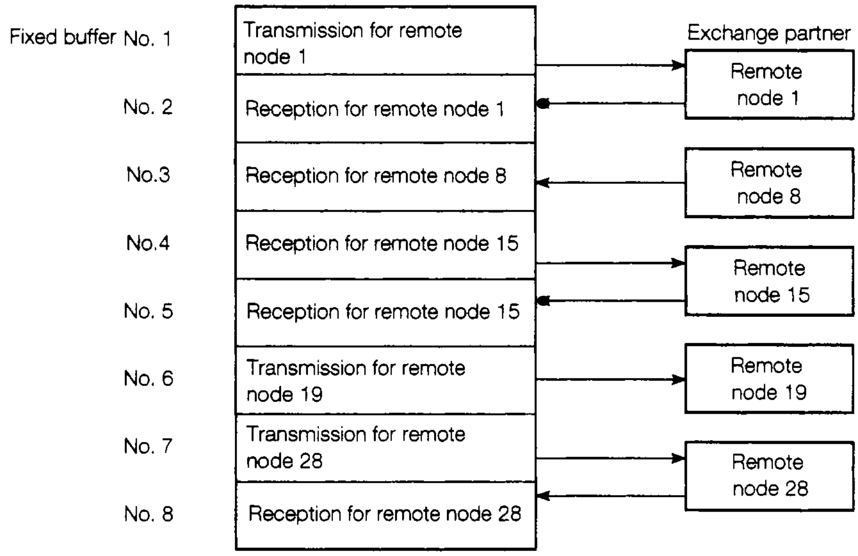
(2) Data exchange can be conducted for remote nodes connected in the Ethernet by the E71, and for remote nodes connected by router relay functions (Refer to Chapter 12).

As shown in the diagram on the following page, the fixed buffers (No.1 to No.8) are used to set the remote terminals to which exchange will be conducted and the usage availability (for transmission/reception, without procedure/with procedure, etc.) to be opened in the E71's communication line (Refer to Item 5.5), and to set the exchange partners for the buffers.

① The parameter settings fixed buffer exchange partner setting when TCP/IP is used, is valid when the E71's open end signal turns from OFF to ON during boot up. The exchange partner cannot be changed while the open end signal is ON.

② The fixed filed exchange partner can be changed after open processing when UDP/IP is used. (The exchange parameter's remote node IP address and remote node port No. can be changed, but the local station E71's port No. cannot be changed.)

(Example)



Point

① Select without procedure and during open the subject connection will be changed to special use for fixed buffer transmission/reception without procedure. (Refer to Item 5.1 (*1))
Fixed buffer exchange with procedure, random access buffer exchange, and reading and writing data to the PLC CPU exchange cannot be conducted at the same time as fixed buffer exchange without procedure.

② When changing the exchange partner, do not perform pairing setting (Refer to Item 5.4.1 $\boxed{1}$ (b) ③), or existence check setting (Refer to Item 5.3.1 *2).
If these are set, the E71 will not operate normally.

(3) The transmission and reception processing during data transmission and reception is as follows.

① During transmission

When the transmission request signal (Y0 to Y7) is ON, the E71 transmits the subject fixed buffer's data to the remote node set in the buffer memory address 18H to 4FH (24 to 79) subject area. (*1)

② During reception

If there is reception from a remote node set in the buffer memory address 18H to 4FH (24 to 79) subject area, the E71 will perform reception processing. (*1) In addition, when reception data is stored in the subject fixed buffer by reception processing, the E71 updates the buffer memory address 59H to A8H (89 to 168) subject connection's remote node IP address and remote node port No.

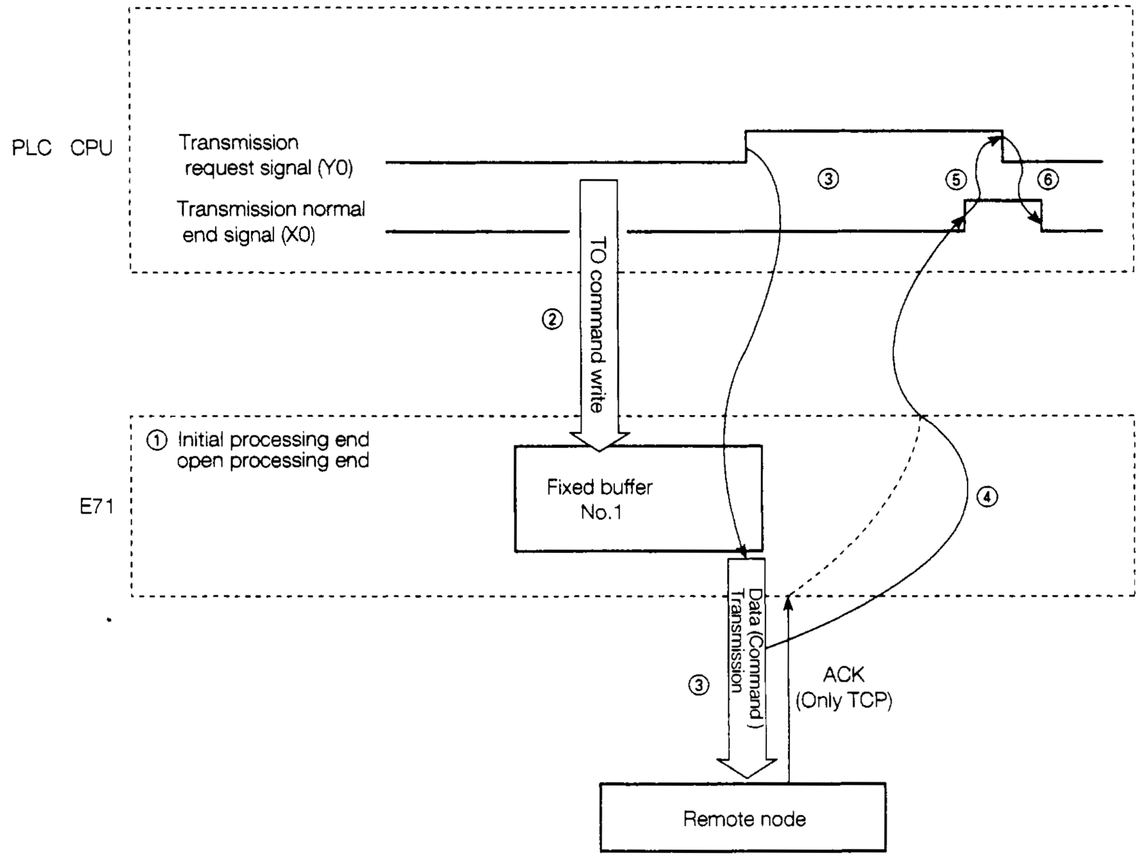
If there is a reception from remote nodes that are not set in the buffer memory address 18H to 4FH (24 to 79), the E71 will ignore the reception data.

For details refer to Item 7.3.2 remarks.

*1 When the TCP/IP unpassive open, data is transmitted to and received from the remote node stored in the buffer memory address 59H to A8H (89 to 168) subject area.

7.1.1 Transmission Control Method

This section explains the control method for transmitting data to the remote node from the E71 using transmission of the fixed buffer No.1 data to a remote node.



① Performs E71 initial processing. (Refer to Items 5.2 through 5.3)

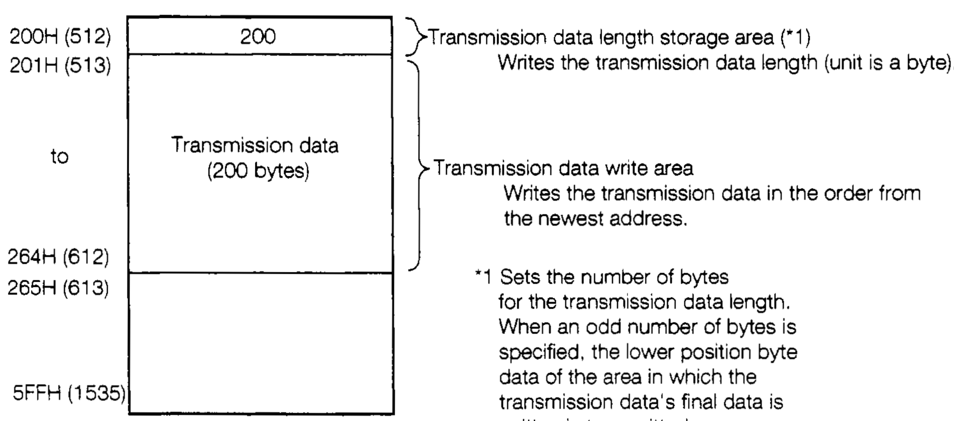
Performs line open processing with the remote node. (Refer to Item 5.4)

② The sequence program's TO command writes the transmission data length and transmission data in the E71's fixed buffer.

The transmission data link is written to the subject fixed buffer's first address (512).

The transmission data is written starting from the subject fixed buffer's first address +1.

The following figure shows an example of a 200-byte transmission using the fixed buffer No.1.



*1 Sets the number of bytes for the transmission data length. When an odd number of bytes is specified, the lower position byte data of the area in which the transmission data's final data is written is transmitted.

③ When the transmission request signal (Y0) is turned ON by the sequence program, the data is transmitted as it is to the specified node (by the parameter settings) from the fixed buffer (No.1).

④ The E71 turns the transmission normal end signal (X0) ON when the data transmission ends.

⑤ The turning ON of the transmission normal end signal causes the sequence program to turn OFF the transmission request signal (Y0).

⑥ When the transmission request signal turns OFF, the transmission normal end signal automatically turns OFF.

Point

(1) When communicating by UDP/IP, need attention for the items described below.
① When the PLC CPU and the remote node are not connected by a communication line due to disconnection of the connection cable, the transmission normal end signal (X0...) and transmission error detection signal (X1...) are not turned ON if data was transmitted to the remote node by the PLC CPU.
② When data is transmitted to the remote node from the PLC CPU, conduct the time out check until transmission is ended using the PLC CPU. (The user can adjust the time out time) When a time out occurs, turn OFF the transmission request signal (Y0...), check the communication line with the partner remote node, and conduct connection processing for the troubled location.

(2) Except (1) in above, when the transmission does not end normally, the transmission error detection signal (X1) turns ON. (The exchange normal end signal (X0) does not turn ON.)
In this case, reconduct transmission processing by turning the transmission request signal from off to on after the transmission error processing is completed.

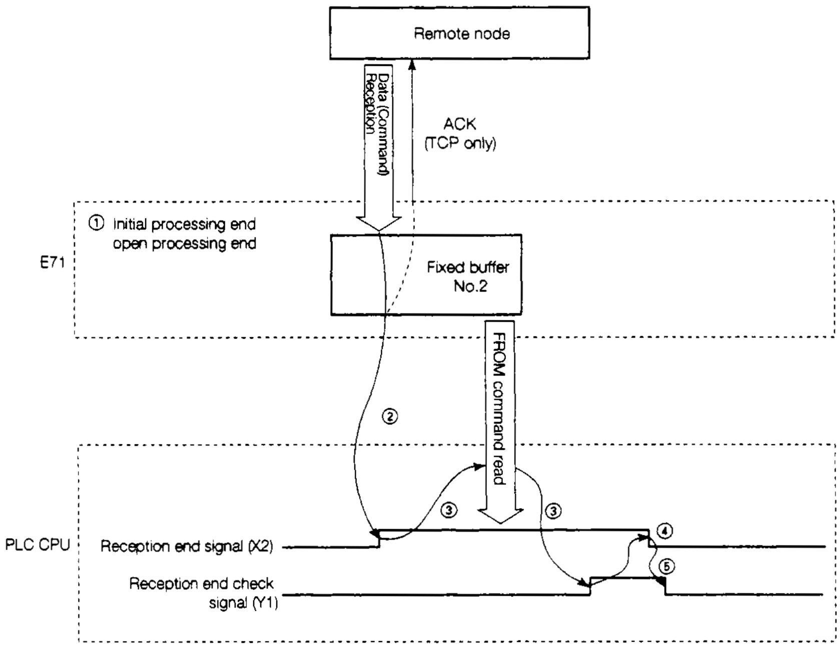
(3) This shows the processing when the open request signal and initial request signal are turned off during transmission when the E71 "Function for Data exchange during the PLC CPU is Stopped"(Item 5.6) is not used.
① When the open request signal (Y8) turns off during transmission, the E71 conducts closed processing after the transmission processing end.
② When the initial request signal (Y19) turns off during transmission, the E71 conducts closed processing and end processing after transmission processing end.

Remarks

For more details on cases when the reception of Close/Abort (RST) has occurred from the partner remote node prior to the completion of the data communication performed immediately before, refer to Remarks in 2 of Item 5.4.3.

7.1.2 Reception Control Method

This section explains the control method that the E71 receives the data from the remote node receiving to the fixed buffer No.2 as an example.



① Conducts E71 initial processing. (Refer to Items 5.2 through 5.3)

Conducts line open processing to the remote node. (Refer to Item 5.4)

To conduct fixed buffer exchange, initial processing and open processing must be completed.

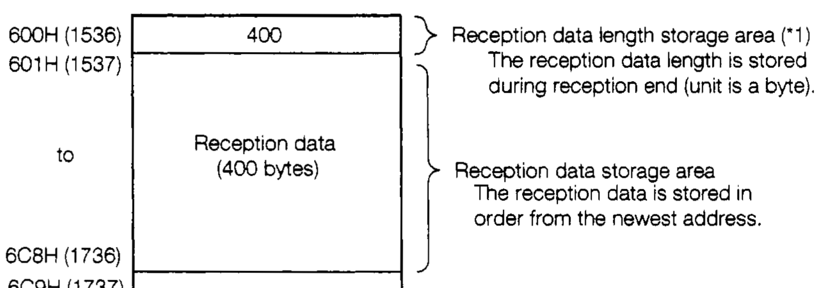
② The E71 will turn on the receive end signal (X2) when the reception data is stored as is from the remote node by parameter setting into the fixed buffer (No.2).

The reception data length and reception data are stored in the fixed buffer.

The reception data length is stored in the subject fixed buffer first address (1536).

The reception data is stored in order starting from the fixed buffers No.1 to 8 first address +1.

The following figure shows an example of a 400-byte reception using fixed buffer No.2.



*1 The reception data length is the number of bytes. When data of an odd number of bytes is received,the last received data is stored in the final storage area's lower position. (The upper position byte is 00H.)

③ The received data length and received data stored in the fixed buffer is read by the sequence program's FROM command when the reception end signal turns on. At the same time, the sequence program turns on the reception end check signal (Y1).

④ The E71 automatically turns off the reception end signal.

⑤ The sequence program turns off the reception end check signal when the reception end signal turns off.

Point

(1) The reception end signal (X2) does not turn on during error data reception. In addition, the data is not stored in the fixed buffer No.2.

(2) This shows the processing when the open request signal and initial request signal are turned off during reception when the E71 "Function for Data Exchange during the PLC CPU is Stopped" (Item 5.6) is not used.

① If the open request signal (Y9) turns off during reception, the E71 immediately conducts close processing.

② If the initial request signal (Y19) turns off during reception, the E71 immediately conducts close processing and end processing.

Remarks

For more details on cases when the reception of Close/Abort (RST) has occurred from the partner remote node prior to the completion of the data communication performed immediately before, refer to Remarks in 2 of Item 5.4.3.

7.2 Data Format

Following shows the exchange data (command) data item order and contents for when fixed buffer exchange is performed without procedure between the E71 and a remote node.

As shown below, the exchange data consists of a header and application data.

7.2.1 Format during Exchange

Following shows the command data item order when fixed buffer communication without procedure is conducted.

1 Transmission/reception data order when exchanging with TCP/IP

Header			Application data
Ethernet	IP	TCP	Text (command)
(14 bytes)	(20 bytes)	(20 bytes)	(Maximum 2046 bytes)

2 Transmission/reception order when exchanging with UDP/IP

Header			Application data
Ethernet	IP	UDP	Text (command)
(14 bytes)	(20 bytes)	(8 bytes)	(Maximum 2046 bytes)

7.2.2 Exchange Data Item Contents

Following shows the command data item contents when conducting fixed buffer exchange without procedures.

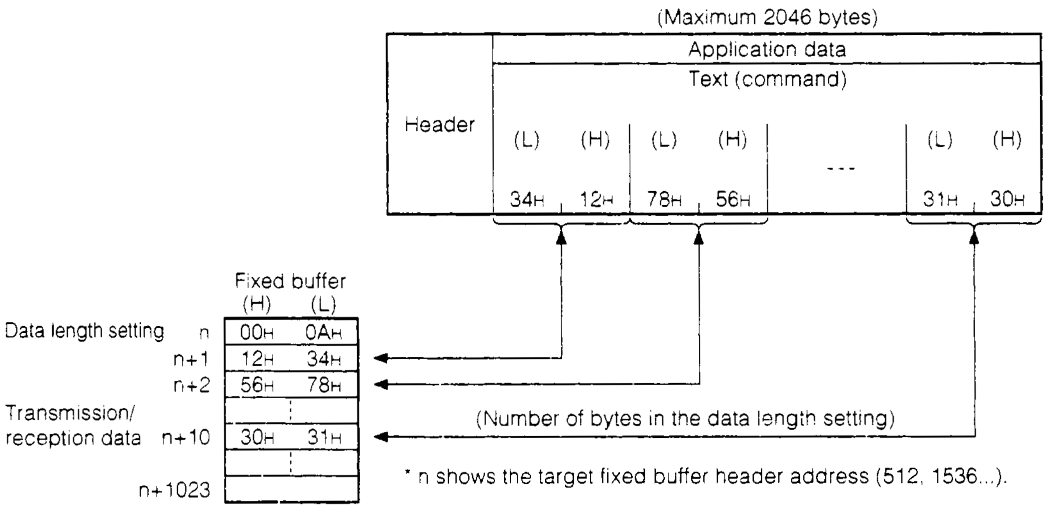
1 Header

The header is the header used for TCP/IP or UDP/IP. For the E71, since the E71 is added or removed the user is not required to make the setting.

2 Text (command)

The text (command) shows the data that is sent to the exchange partner node. The data code is expressed in binary code. The binary and ASCII setting made using the DIP switch on the front of the E71 is ignored.

(Example) Text (command) portion format during exchange



Remarks

This subheader and data length added to the fixed buffer exchange with procedures does not exist when without procedure. All the data is handled as valid text.

7.3 Simultaneous Broadcast Communication When Using UDP/IP

When UDP/IP is used to conduct fixed buffer exchange without procedure, a simultaneous broadcast to all E71 installation stations connected to the same Ethernet as the E71.

This makes it possible to write, etc., the same data to all stations.

Point

(1) For simultaneous broadcast communication, the necessity of the reception message must be determined, and if unnecessary, read and discard processing must be conducted for remote nodes connected to the same Ethernet.

(2) For simultaneous broadcast communication, the user arranges the simultaneous broadcast transmission, reception special port No., and specifies the port No.

7.3.1 Simultaneous Broadcast Communication Transmission

When conducting open processing for the partner remote node IP address to which data will be sent as FFFFFFFFH, simultaneous broadcast communication transmission can be conducted. During simultaneous broadcast communication transmission, the E71 changes the request destination IP address to FFFFFFFFH, and transmits the data on the Ethernet.

(Example) When connection 1 is used

Address	Buffer memory	
10H(16)	0300H	Set without procedure, UDP, and for transmission.
to	to	
18H(24)	E71 Port No.	
19H(25)	FFFFH	Makes the remote node IP address to a simultaneous broadcast address.
1AH(26)	FFFFH	
1BH(27)	Remote node port No.	Makes the remote node port No. the simultaneous broadcast port No. (arranged by the user)

During the situation shown in the figure above, the E71 makes the request destination IP addresses equal FF. FF. FF. FFH, makes the request destination port No. equal the remote node port No., and transmits the fixed buffer's data.

7.3.2 Simultaneous Broadcast Communication Reception

Making the partner remote nodes IP address to which the data will be received FFFFFFFFH and the port No. to FFFFH / User arrangement No. and conducting open processing will receive process all of the corresponding reception data as simultaneous broadcast communication data.

(Example) When connection 1 is used

Address	Buffer memory	
10H(16)	0301H	Set without procedure, UDP, and for Reception
to	to	
18H(24)	E71 Port No.	
19H(25)	FFFFH	Makes the remote node IP address to a simultaneous broadcast address.
1AH(26)	FFFFH	
1BH(27)	FFFFH	Makes the remote node port No. the simultaneous broadcast port No. (FFFFH/ User arrangement No.)

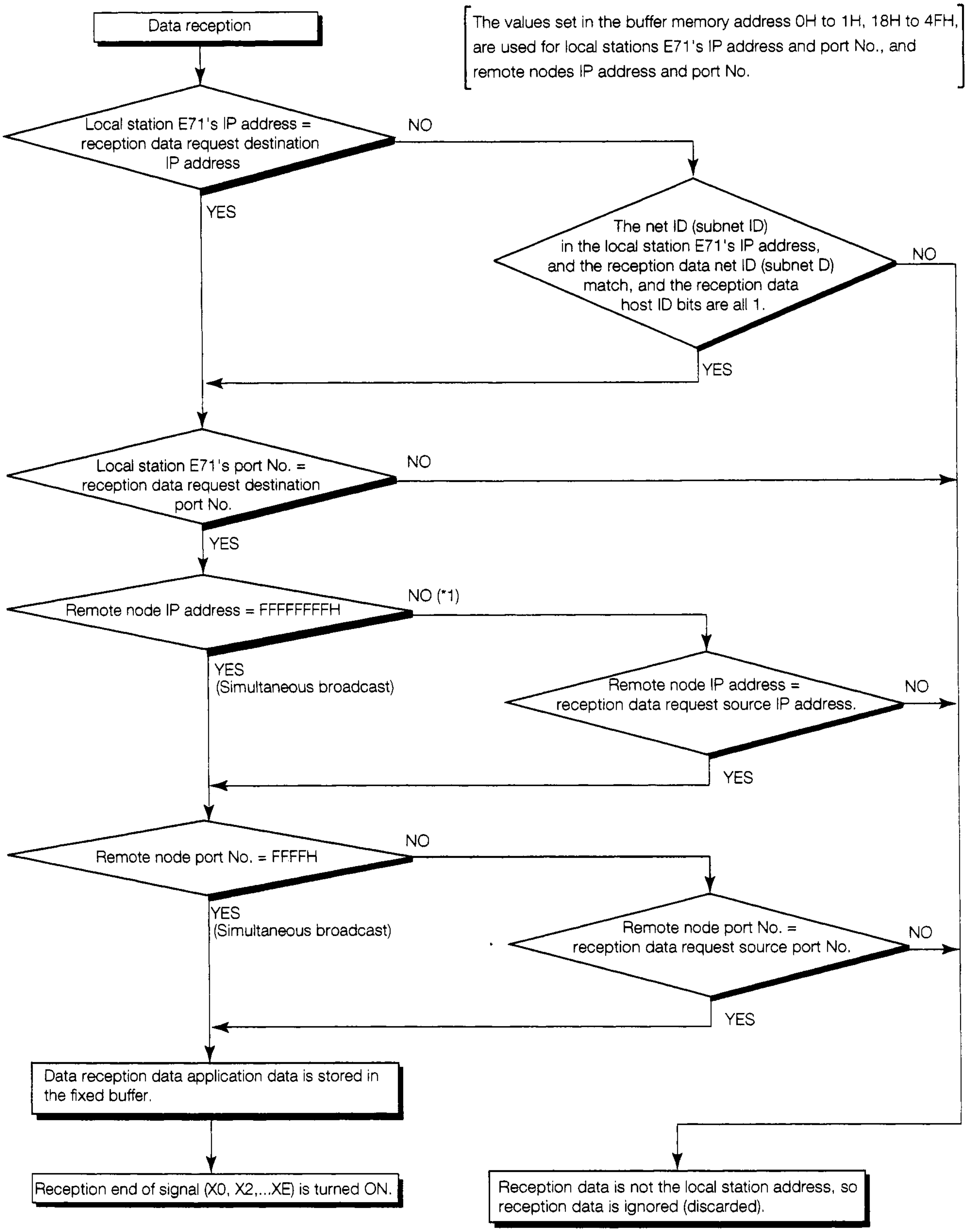
During the situation shown in the above figure, the E71 changes all of the bits in the reception data request destination IP addresses local station class host ID range to 1, and if the request destination port No. equals the E71's port No., the reception data is stored in the fixed buffer and the reception end signal (X0) is turned on.

In addition, when the reception data is stored in the subject fixed buffer, the E71 updates the buffer memory address 59H to A8H (89-168) of the subject area's remote node IP address and the remote node's port No.

When checking the data transmission origin, read the above buffer memory (exchange state storage areas information area by connection).

Remarks

Following shows an overview of the E71's internal processing when there is reception using without procedure and reception using simultaneous broadcast communication.



*1 When all of the bits in the range that represents the reception data request destination IP address' host ID are 1, processing is conducted on the yes side.

7.3.3 Precautions When Using Simultaneous Broadcast Communications Functions

Following are precautions when conducting simultaneous broadcast communication with fixed buffer exchange without procedures.

1 **For simultaneous broadcast communication, the user will arrange the simultaneous broadcast transmission/reception special port No. and specify the port No. for it.**

2 **The simultaneous broadcast communication transmission message is set to all the nodes on the Ethernet to which the E71 is connected.**

All of the nodes connected to the same Ethernet determine whether the received message broadcast by simultaneous broadcast communication is necessary, and when unnecessary, must conduct read and discard processing.

* When the transmitted message is unrelated to a particular node, the particular node will discard the received message. In addition, even if a particular node is the subject station, do not return a response. The E71 will automatically perform this processing.

3 **The application data amount that can be handled at one time for transmission or reception is a maximum of 2046 bytes. If data of 2047 bytes or more must be transmitted or received, divide it at the transmission origin.**

4 **When the reception data is read to the PLC CPU when the reception end signal (X0, X2...XE) is turned on, be sure to turn on the corresponding reception end check signal (Y0 to Y7).**

Turning on the reception end check signal (Y0 to Y7) makes it possible for the E71 to store the received data in the subject fixed buffer.

Not turning on the reception end check signal (Y0 to Y7) could cause the reception data to be discarded.

Remark

During the period between when the reception end signal turns on to when the reception end check signal is turned on, the data to be received (including the header) is stored in the E71's OS internal buffer.

The OS internal buffer capacity is approximately 43k bytes. The portion of the reception data that exceeds this capacity will be discarded.

7.4 Programming

This section explains the programming method for conducting exchange between the E71 and the remote node using the fixed buffer and without procedure.

7.4.1 Precautions When Creating Programs

1 **Fixed buffer exchange can be conducted when the open request signal(Y8 to YF) and the open end signal (X10 to X17) are turned on. The initial processing and communication line open processing must be completed. (Refer to Chapter 5)**

2 **The parameter setting contents are taken into the E71 when the open request signal (Y8 to YF) is turned from off to on at the boot up.**

Except for that in 3 below, the control contents cannot be changed even if the parameter contents are written over while the open end signal (X10 to X17) is on.

3 **When using a connection that is UDP open,**

(a) The settings values of the exchange parameter setting area's exchange address setting area can be changed before transmission or reception, and the exchange partner remote node can be changed.

Therefore, data can be transmitted in order to multiple nodes, so conduct transmission and reception by switching the partner remote nodes to prevent exchange trouble from occurring.

(b) When transmission data, need attention for the items described below.

① When the PLC CPU and the remote node are not connected by a communication line due to disconnection of the connection cable, the transmission normal end signal (X0...) and transmission error detection signal (X1...) are not turned ON if data was transmitted to the remote node by the PLC CPU.

② When data is transmitted to the remote node from the PLC CPU, conduct the time out check until transmission is ended using the PLC CPU. (The user can adjust the time out time) When a time out occurs, turn OFF the transmission request signal (Y0...), check the communication line with the partner remote node, and conduct connection processing for the troubled location.

4 **When opened by selecting without procedure, the subject connection becomes a fixed buffer transmission/reception special use without procedure, so fixed buffer exchange with procedure, random access buffer exchange, and reading and writing data to the PLC CPU exchange, cannot be conducted at the same time as fixed buffer exchange without procedure.**

5 **The data length specified (stored) in the buffer memory during exchange without procedure is byte units. (The data length during exchange with procedure is in word units.)**

When the buffer memory transmission data length exceeds the range during data transmission, an exchange error will occur and transmission will not be conducted.

6 **When receiving data using the fixed buffer, be sure to turn on the reception end check signal (Y0 to Y7) during reception end (at the point the reception end signal turns on).**

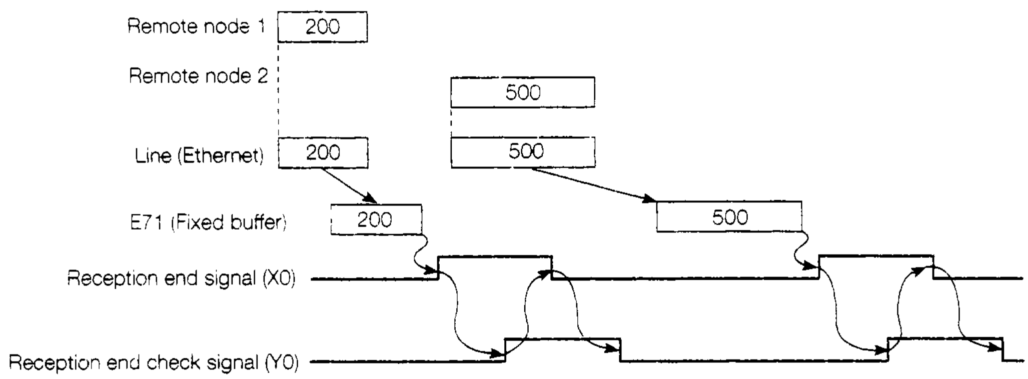
Turning on the reception end check signal stores the following reception data in the subject fixed buffer.

7 When using without procedure, the message does not have a data length.

The E71 turns on the reception end signal (X0, X2,... XE) after the received message (packet) size is stored in the reception data length storage area.

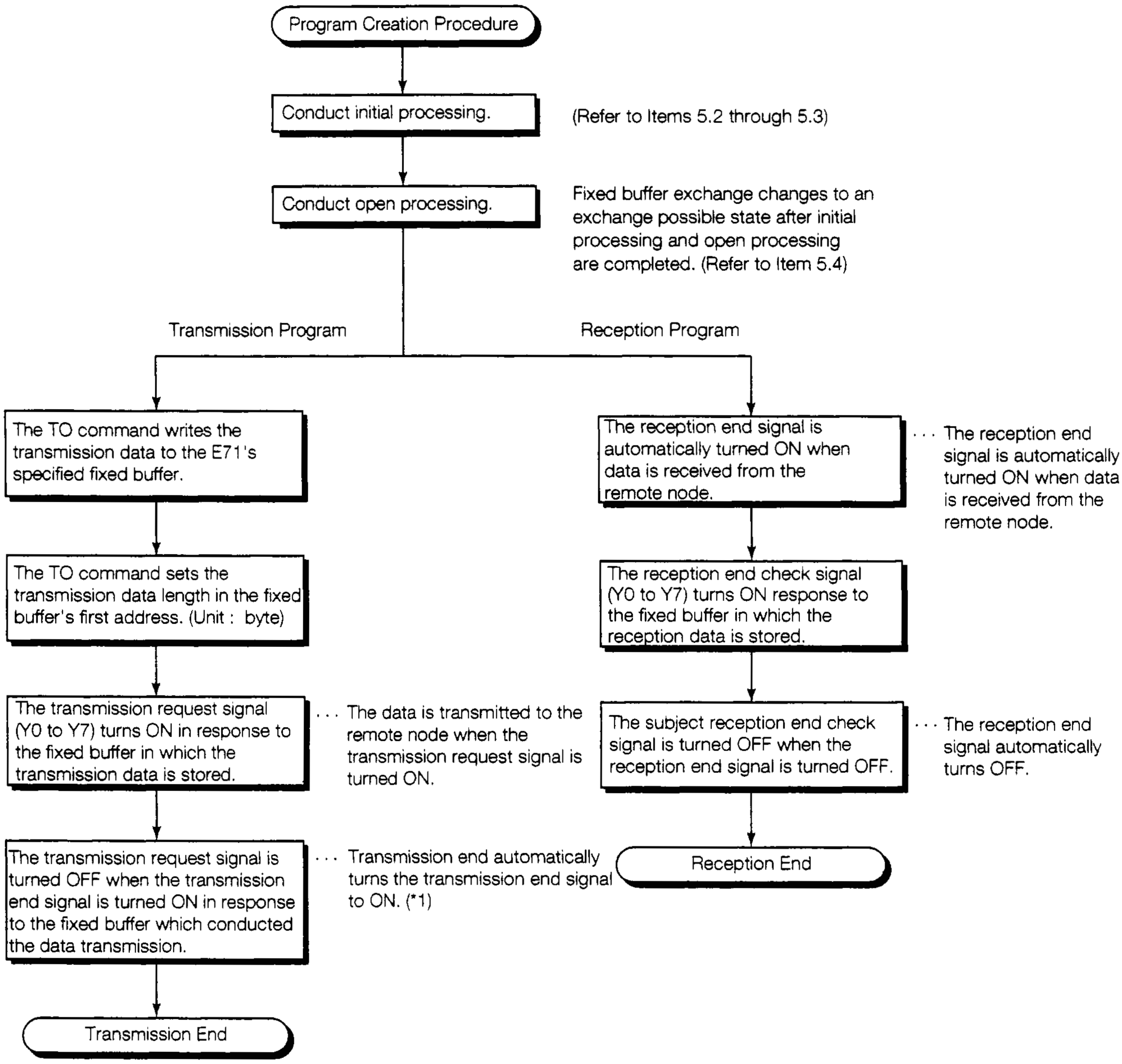
It is recommended that check steps, such as including data length and data type codes in the message application data, be used to allow the receiving end to identify the number of application data bytes and the data types.

(Example) For continuous reception of messages from remote nodes 1 and 2.



7.4.2 Program Creation Procedures

This section explains the data transmission and reception program creation procedures using fixed buffer without procedure.



*1 When the transmission error detection signal is on, handle the exchange state storage areas information for individual connection (transmission error code) as specified in Chapter 13.

7.4.3 Example Fixed Buffer Exchange Program (Without Procedure)

This section explains the programming method for performing data exchange with a remote node using fixed buffer No.1 with procedure.

(Program Conditions)

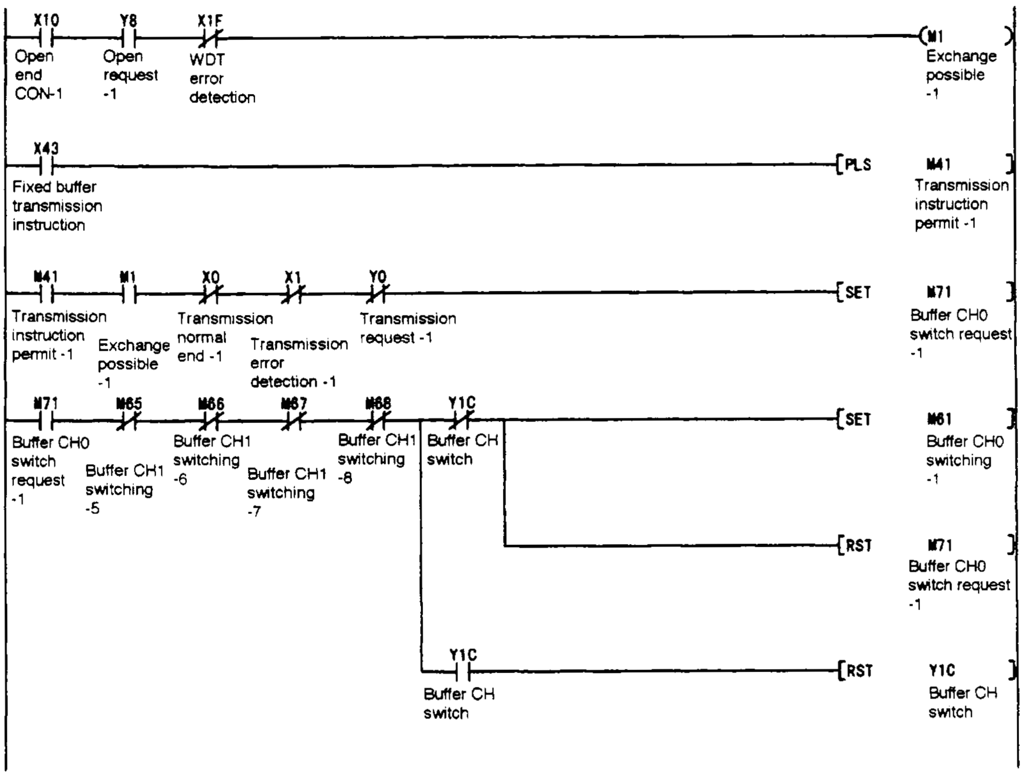
(a) Set the exchange parameters for each connection as specified in Item 5.4.5.

(b) Set the fixed buffer transmission data to D300 to D399.

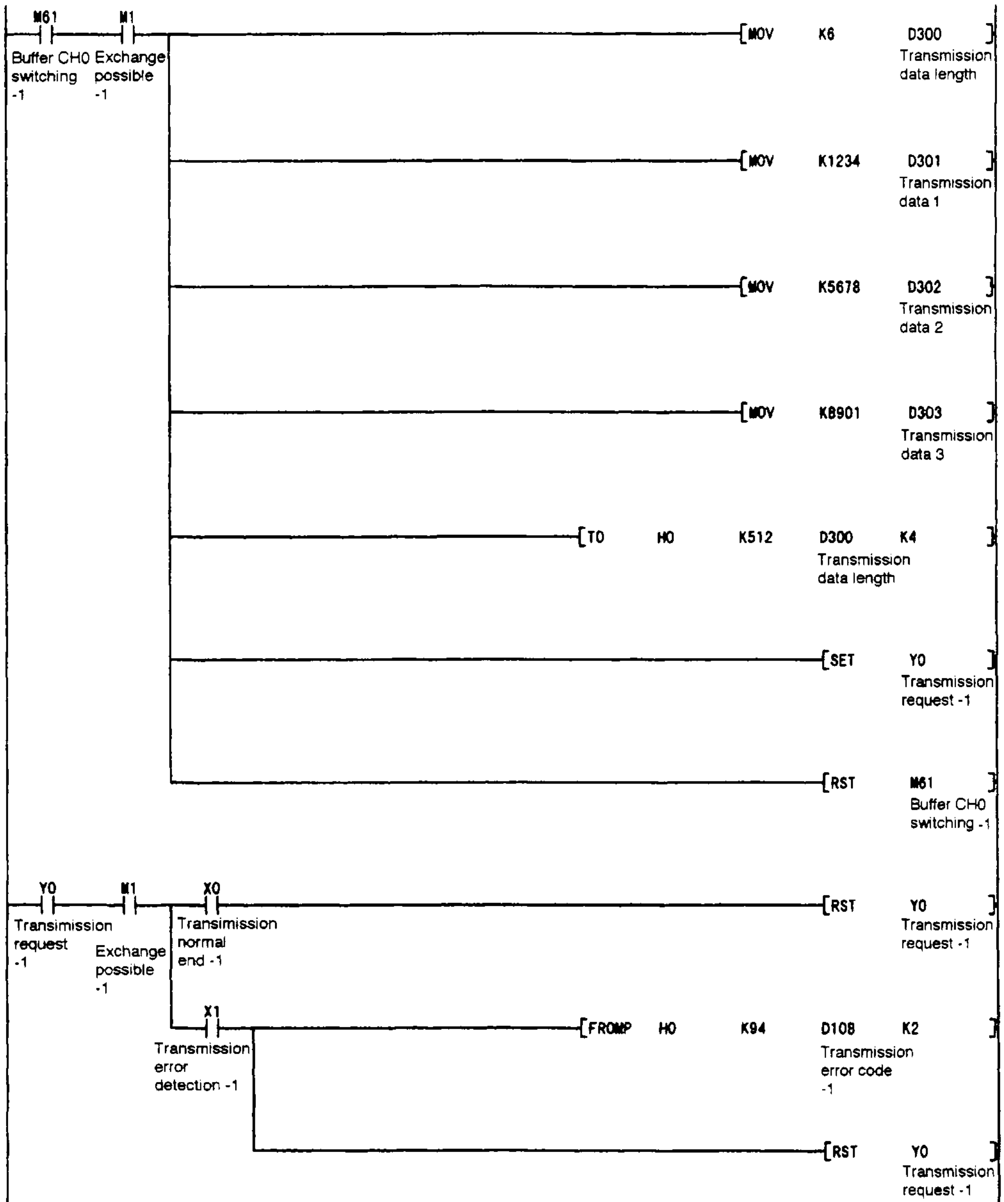
(c) Store the fixed buffer reception data in D500 to D599.

(d) The storage destination for the error code and end code are allotted as follows.
D108 Transmission error code
D109 Exchange end code
D110 Reception error code

Fixed Buffer No.1 Transmission Program



Channel switch for buffer memory completed. Writing to the fixed buffer of connection 1 is possible.



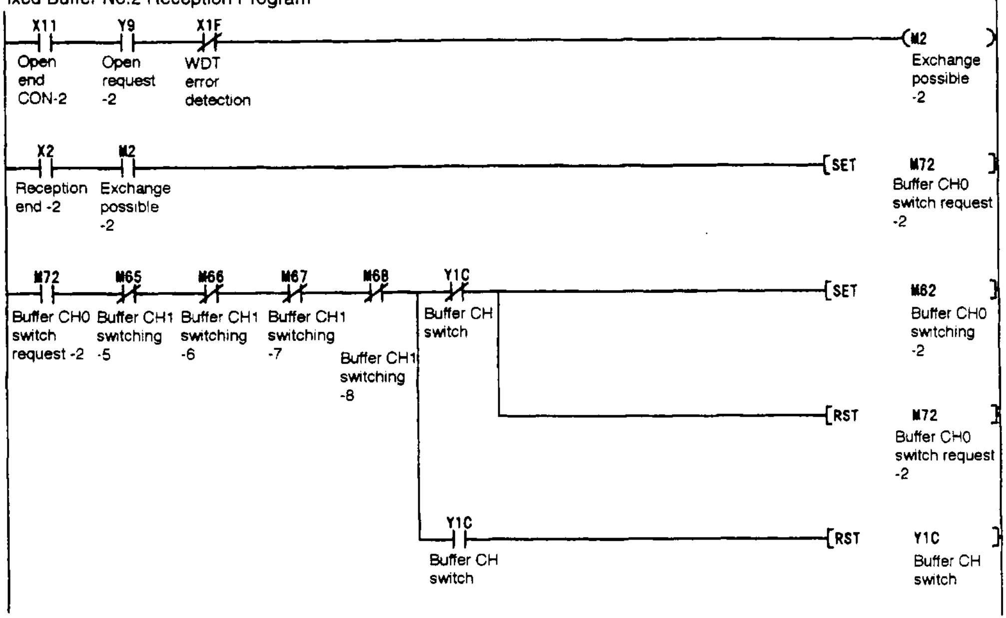
Set the data length to 6 bytes.

Set the transmission data.

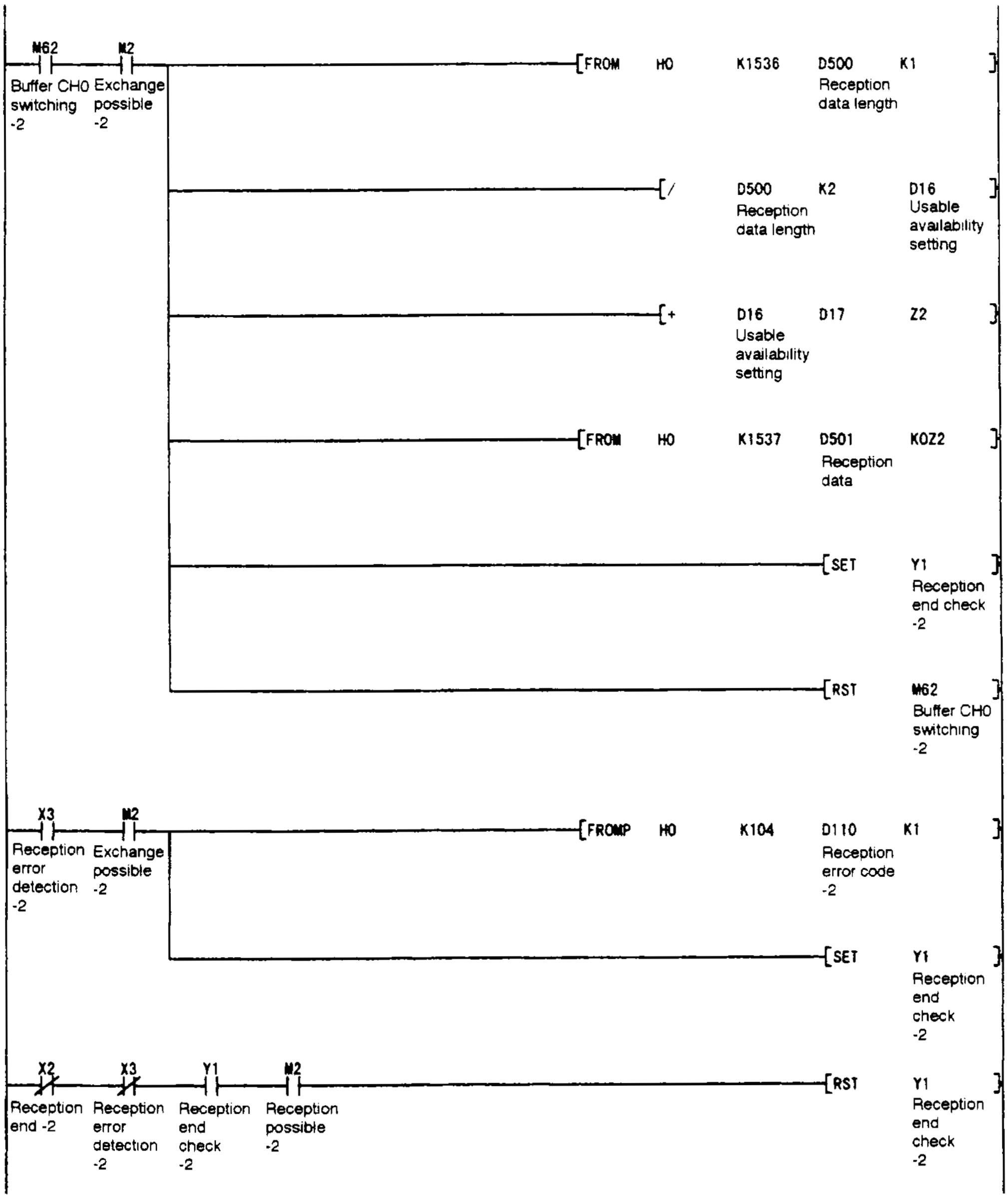
Writes the data length and transmission data in the fixed buffer No.1's area.

Reads an error information when a transmission error occurs.

Fixed Buffer No.2 Reception Program



Channel switch for buffer memory completed.
Reading from the fixed buffer of connection 2 is possible.



The reception data is read to D501 to when the end of the data length is reached when the reception end signal is turned ON.

Reads an error code when a reception error occurs.

- When the I/O control method of the PLC CPU of the station installed in the E71 is the refresh method and a fixed buffer read/write is performed after the buffer memory channel switching signal (Y001C) is switched from on → off/off → on, a read/write must be performed after the channel switching signal (Y001C) is output to the E71. Output the channel switching signal (Y001C) to the E71 using the tail's *1 of Item 3.6.2.

RANDOM ACCESS BUFFER EXCHANGE SECTION

The random access buffer exchange section explains the method for exchanging data between the remote node's external equipment and the PLC CPU using the Ethernet interface module's random access buffer.

Random access buffer exchange begins after the initial processing and open processing that connects the communication line as described in Chapter 5.

In addition, perform close processing and end processing when data exchange is completed for the subject communication line.

8. RANDOM ACCESS BUFFER EXCHANGE

This section explains the method for conducting exchange with remote nodes using the E71's random access buffer.

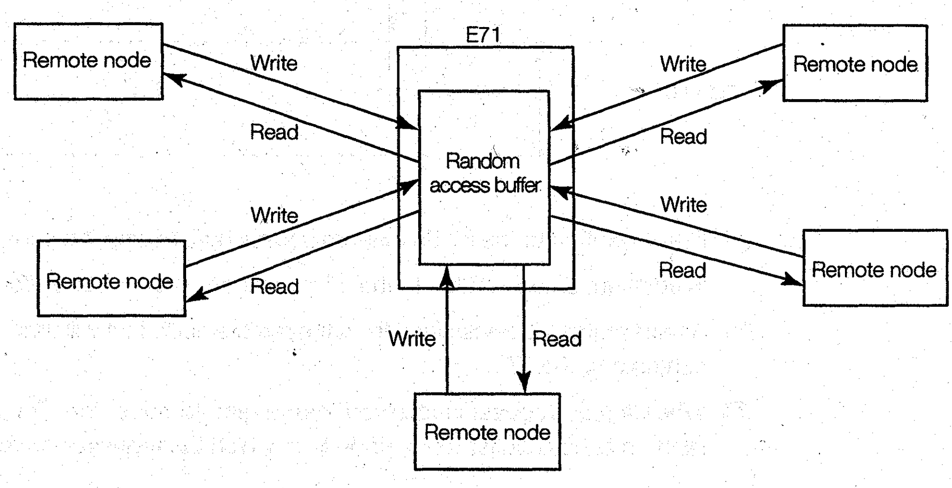
8.1 Control Format

This section explains the exchange processing control format using the random access buffer.

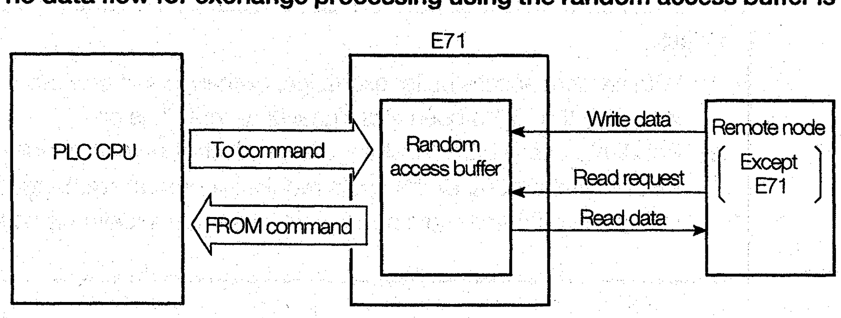
For random access buffer exchange, data is written to the random access buffers and read from the random access buffer using commands (requests) from the remote nodes.

Writing to and reading from the E71's random access buffer from the remote nodes is conducted asynchronously with the PLC CPU's sequence program.

1 **The random access buffer is not fixed to the remote node to which exchange is done, but writing and reading can be freely conducted from any remote node (except E71). Therefore, a common buffer area can be used for all nodes connected to the Ethernet.**



2 **The data flow for exchange processing using the random access buffer is shown below.**



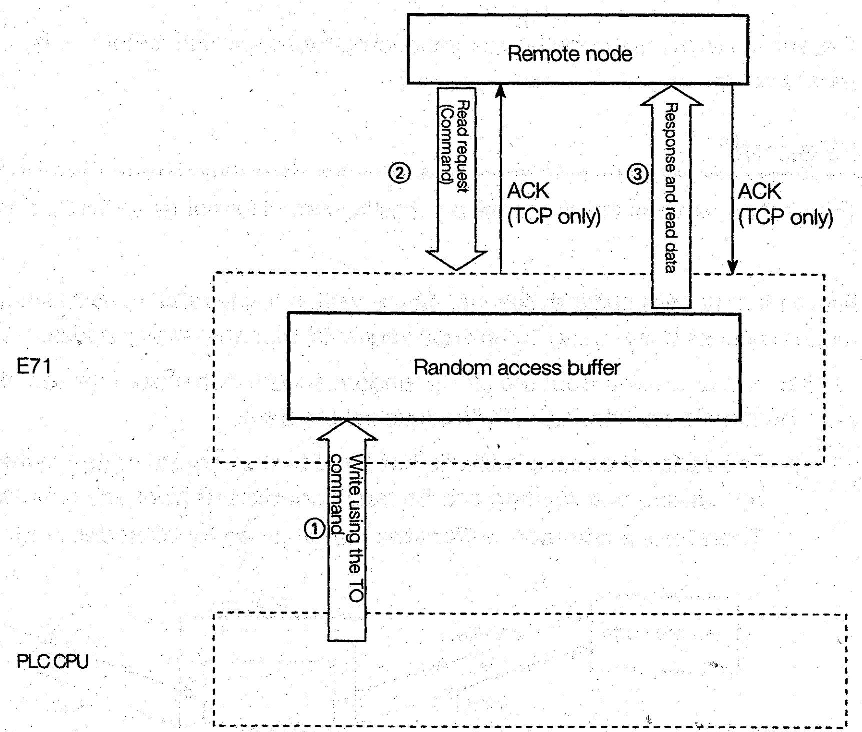
3 **Random access buffer exchange can be conducted from the remote node shown below except for E71, AJ71E71, and QE71. (Exchange cannot be done between E71 and E71, AJ71E71, or QE71.)**

① Remote nodes on the Ethernet to which the E71 is connected.

② Remote nodes on the Ethernet that are connected using router relay functions (Refer to Chapter 12).

8.1.1 Control Method When there is a Read Request from a Remote Node

This section explains the control method when data is transmitted from the E71 by a read request from a remote node.



① Data is written to the E71's random access buffer using the sequence program's TO command. In addition, data is written to the E71's random access buffer from a remote node.

② A read request is transmitted from the remote node that will read the E71's random access buffer contents to the E71.

③ When a read request is received from a remote node, the E71 will send the data written in the random access buffer to the node from which the request was received as a response.

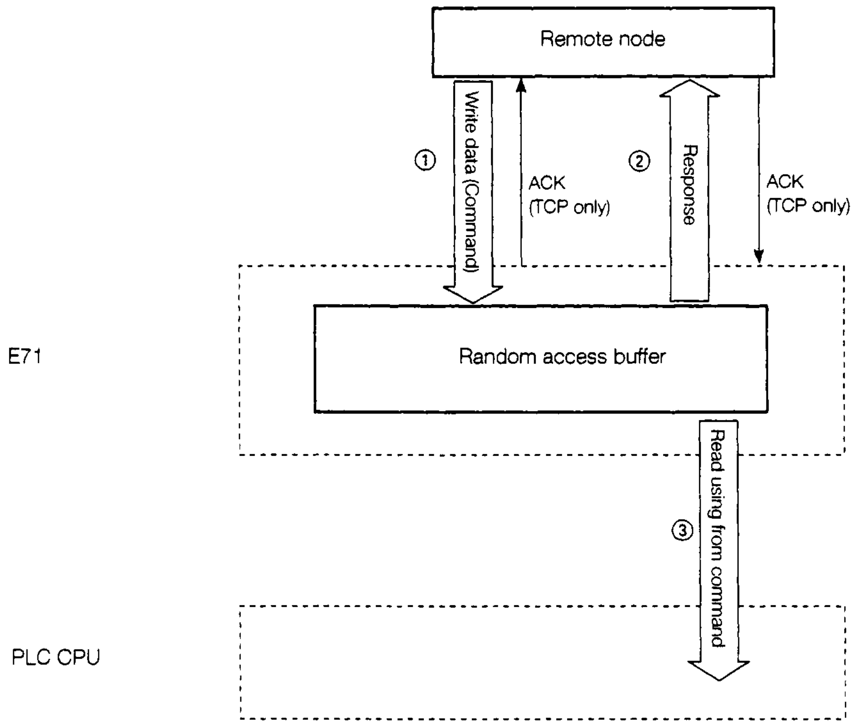
Point

(1) With random access buffer exchange, exchange can only be conducted with the remote node for which the E71's open end signal (X10 to X17) is on.

(2) Random access buffer exchange is conducted asynchronously with the sequence program. When synchronous exchange is required, conduct exchange by putting a free protocol between the partner remote node to which communication is being done and the PLC CPU.

8.1.2 Control Method When there is a Write Request from a Remote Node

This section explains the control mode when data is written by the remote node to the E71's random access buffer.



① Data is written from the remote node to the E71's random access buffer.

② When the E71 receives data from the remote node, a response is returned to the remote node that conducted the transmission.

③ The data received by the random access buffer using the sequence program's FROM command is read.

In addition, the data being received to the E71's random access buffer can be read by the separate remote node.

Point

(1) For random access buffer exchange, exchange can only be conducted with remote nodes for which the E71's open end signal (X10 to X17) is on.

(2) Random access buffer exchange is conducted asynchronously with the sequence program. When synchronous is required, conduct exchange by adding a free protocol between the partner remote node to which exchange is being conducted and the PLC CPU.

8.2 Data Format

This shows the exchange data (command and response) data item order and contents when random access buffer exchange is conducted between the E71 and a remote node. As shown below, the exchange data consists of a header and application data.

Header	Application data
	Maximum 2044 bytes

As shown below, the data code of the application data can be shown as either binary or ASCII code.

Use the DIP switch on the front of the E71 to set either binary or ASCII. (For details regarding the setting method refer to Item 4.3.2.)

8.2.1 Format When Exchanging with Binary Code

Following shows the command in response data item order when exchanging the exchange data application data portion in binary code data when conducting random access buffer exchange.

1 Transmission/reception data order when exchanging using TCP/IP

(a) When a read request is made from the remote node

① Order during command transmission

Header			Application data		
Ethernet	IP	TCP	Subheader	Head address	Data length setting
				(L) (H)	(L) (H)
			61_H, 00_H		
(14 bytes)	(20 bytes)	(20 bytes)	(2 bytes)	(2 bytes)	(2 bytes)

② Order during response reception

Header			Application data		
Ethernet	IP	TCP	Subheader	End code	Text (command)
			$E1_H$		
(14 bytes)	(20 bytes)	(20 bytes)	(1 byte)	(1 byte)	(Maximum 1017 words)

(b) When a write request is received from remote node

① Order during command transmission

Header			Application data			
Ethernet	IP	TCP	Subheader	Head address	Data length setting	Text (command)
				(L) (H)	(L) (H)	
			62_H, 00_H			
(14 bytes)	(20 bytes)	(20 bytes)	(2 bytes)	(2 bytes)	(2 bytes)	(Maximum 1017 words)

② Order during response reception

Header			Application data	
Ethernet	IP	TCP	Subheader	End code
			$E2_H$	
(14 bytes)	(20 bytes)	(20 bytes)	(1 byte)	(1 byte)

2 Transmission/reception data order when exchanging using UDP/IP

(a) When a read request is made from the remote node

① Order during command transmission

Header			Application data		
Ethernet	IP	UDP	Subheader	Head address	Data length setting
			61H, 00H	(L) (H)	(L) (H)
(14 bytes)	(20 bytes)	(8 bytes)	(2 bytes)	(2 bytes)	(2 bytes)

② Order during response reception

Header			Application data		
Ethernet	IP	UDP	Subheader	End code	Text (command)
			E1H		
(14 bytes)	(20 bytes)	(8 bytes)	(1 byte)	(1 byte)	(Maximum 1017 words)

(b) When a write request is received from remote node

① Order during command transmission

Header			Application data			
Ethernet	IP	UDP	Subheader	Head address	Data length setting	Text (command)
			62H, 00H	(L) (H)	(L) (H)	
(14 bytes)	(20 bytes)	(8 bytes)	(2 bytes)	(2 bytes)	(2 bytes)	(Maximum 1017 words)

② Order during response reception

Header			Application data	
Ethernet	IP	UDP	Subheader	End code
			E2H	
(14 bytes)	(20 bytes)	(8 bytes)	(1 byte)	(1 byte)

8.2.2 Format When Exchanging with ASCII Code

The following shows the command in response data item order when exchanging the exchange data application data portion in ASCII code data when conducting random access buffer exchange.

1 Transmission/reception data order when exchanging using TCP/IP

(a) When a read request is made from the remote node

① Order during command transmission

Header			Application data		
Ethernet	IP	TCP	Subheader	Head address	Data length setting
			"6" "1" "0" "0" 36H 31H 30H 30H	(H) (L)	(H) (L)
(14 bytes)	(20 bytes)	(20 bytes)	(4 bytes)	(4 bytes)	(4 bytes)

② Order during response reception

Header			Application data		
Ethernet	IP	TCP	Subheader	End code	Text (command)
			"E" "1" 45H 31H		
(14 bytes)	(20 bytes)	(20 bytes)	(2 bytes)	(2 bytes)	(Maximum 1016 words)

(b) When a write request is received from remote node

① Order during command transmission

Header			Application data			
Ethernet	IP	TCP	Subheader	Head address	Data length setting	Text (command)
			"6" "2" "0" "0" 36H 32H 30H 30H	(H) (L)	(H) (L)	
(14 bytes)	(20 bytes)	(20 bytes)	(4 bytes)	(4 bytes)	(4 bytes)	(Maximum 1016 words)

② Order during response reception

Header			Application data	
Ethernet	IP	TCP	Subheader	End code
			"E" "2" 45H 32H	
(14 bytes)	(20 bytes)	(20 bytes)	(2 bytes)	(2 bytes)

2 Transmission/reception data order when exchanging using UDP/IP

(a) When a read request is made from the remote node

① Order during command transmission

Header			Application data		
Ethernet	IP	UDP	Subheader "6" "1" "0" "0" 36H 31H 30H 30H	Head address (H) (L)	Data length setting (H) (L)
(14 bytes)	(20 bytes)	(8 bytes)	(4 bytes)	(4 bytes)	(4 bytes)

② Order during response reception

Header			Application data		
Ethernet	IP	UDP	Subheader "E" "1" 45H 31H	End code	Text (command)
(14 bytes)	(20 bytes)	(8 bytes)	(2 bytes)	(2 bytes)	(Maximum 1016 words)

(b) When a write request is received from remote node

① Order during command transmission

Header			Application data			
Ethernet	IP	UDP	Subheader "6" "2" "0" "0" 36H 32H 30H 30H	Head address (H) (L)	Data length setting (H) (L)	Text (command)
(14 bytes)	(20 bytes)	(8 bytes)	(4 bytes)	(4 bytes)	(4 bytes)	(Maximum 1016 words)

② Order during response reception

Header			Application data	
Ethernet	IP	UDP	Subheader "E" "2" 45H 32H	End code
(14 bytes)	(20 bytes)	(8 bytes)	(2 bytes)	(2 bytes)

8.2.3 Exchange Data Item Contents

This shows the command in response data item contents when conducting random access buffer exchange.

1 Header

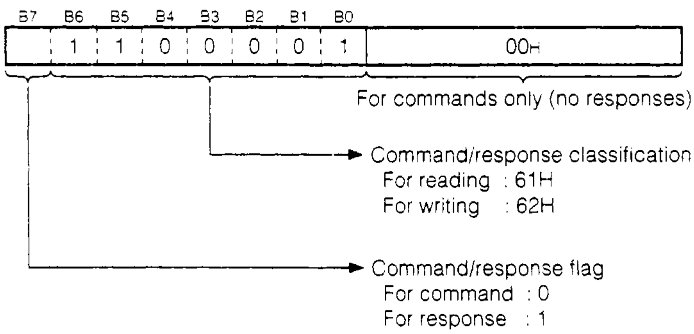
This header is the header that is used for TCP/IP or UDP/IP. For E71, since this can be added or removed the user is not required to make the settings.

2 Subheader

The subheader format is as shown below.

For the E71, since the E71 text is automatically added and removed the user is not required to make the settings. The subheader data code order when conducting random access buffer exchange is shown below.

	Subheader code during exchange			
	For reading		For writing	
	For commands	For response exchange	For commands	For response exchange
For binary code exchange	61H 00H	E1H	62H 00H	E2H
For ASCII code exchange	36H 31H 30H 30H	45H 31H	36H 32H 30H 30H	45H 32H



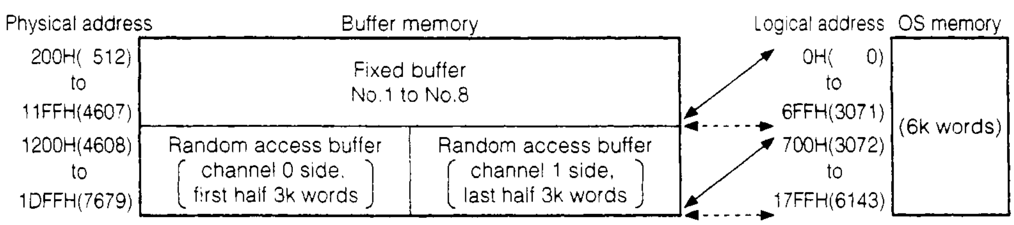
3 Head address

This shows the random access buffer range head address (1200H to 1DFFH) using a logical address (0H to 17FFH, refer to Item 8.3.1) when reading/writing data.

(a) Specifies the head address with a binary value when exchanging with binary code.

(b) Specifies ASCII code when the head address is expressed using a hexadecimal numeral when exchanging with ASCII code.

(c) The random access buffer specification address is as shown below.



Remarks

The random access buffer specified address differs from the address specified by the remote node and the address specified by the PLC program FROM/TO instruction, so practice due caution.

Physical address .... Address specified by the PLC program FROM/TO instruction.

Logical address .... Address specified in the head address in the command for random access buffer exchange.

4 **Data length setting**

This shows the number of read/write data words in the random access buffer range.

(a) When exchanging using binary code the number of words is specified by binary value.

(b) When exchanging using ASCII code, the number of words is expressed in hexadecimal and is specified by ASCII code.

Point

(1) When binary code has been specified the size of the random access buffer subject to read/write is a maximum of 1017 words.

(2) When ASCII code is specified, the size of the random access buffer subject to read/write is a maximum of 508 words. This is approximately one-half of that when binary code is specified.

5 **Text**

This shows the write and read data for the random access buffer.

(a) When exchanging using binary code the data code in the random access buffer is transmitted/received as is.

(b) When exchanging using ASCII code the data in the random access buffer is transmitted/received after being converted to ASCII code.

(c) For information regarding the data code and order when transmitting/receiving data refer to Item 8.2.4.

6 **End code**

Shows the end code added to the response during random access buffer exchange.

The end code is stored in the buffer memory exchange state storage area.

When Binary Code is Specified		When ASCII Code is Specified	
00H	Normal end	30H30H	Normal end
50H	Command/Response type undefined error	35H30H	Command/Response type undefined error
51H	Head address defective	35H31H	Head address defective
52H	Number of data words defective	35H32H	Number of data words defective
————		35H34H	ASCII conversion error

(For details regarding error codes, refer to Chapter 13.)

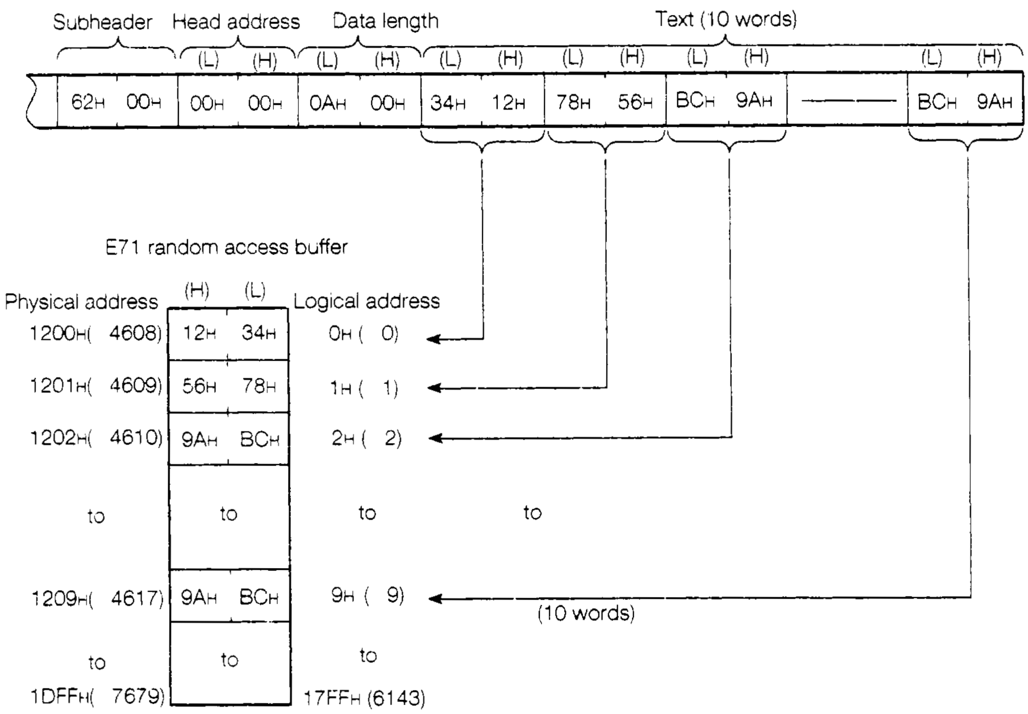
8.2.4 Example Command and Response Format

Following is an example of the command and response format during random access buffer exchange.

1 **Write to buffer by write request from remote node**

When binary code specified

(a) Command format (Remote node → E71)

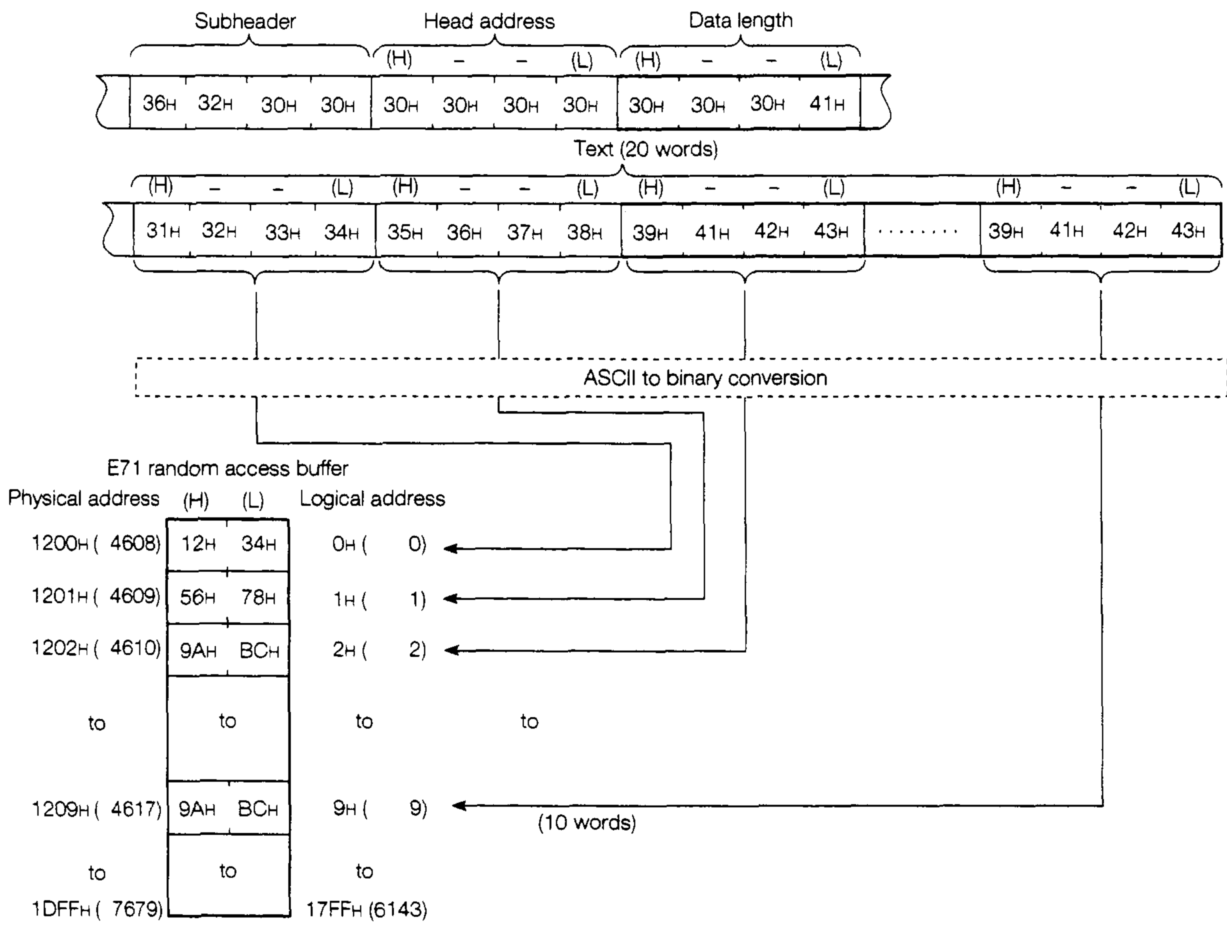


(b) Response format (Remote node ← E71)

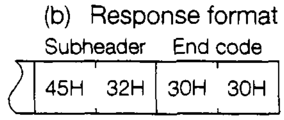
Subheader	End code
E2H	00H

When ASCII code specified

(a) Command format (Remote node → E71)



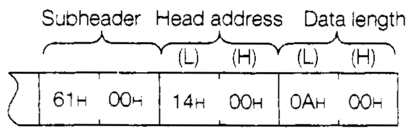
(b) Response format (Remote node ← E71)



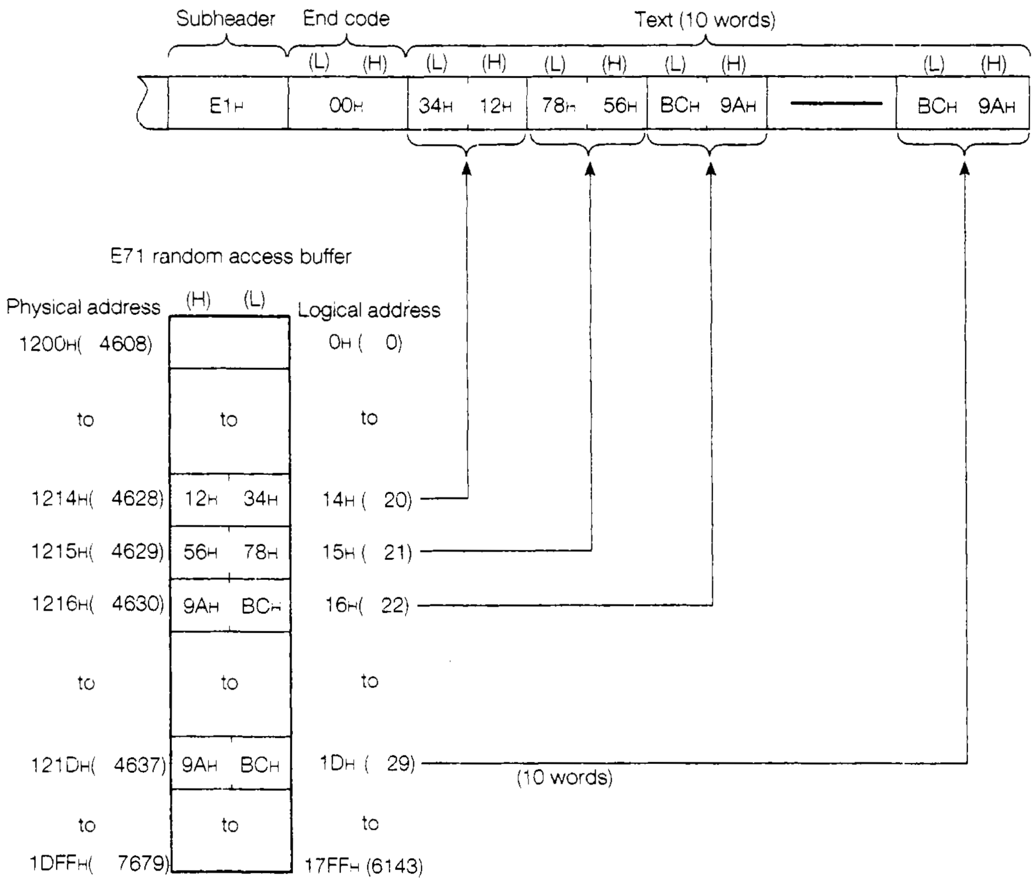
2 **Read from buffer using read request from remote node**

When binary code specified

(a) Command format (Remote node → E71)

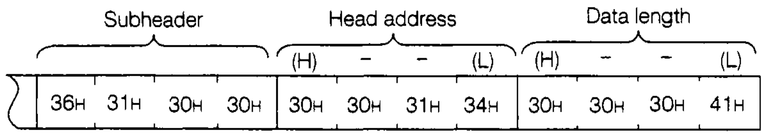


(b) Response format (Remote node ← E71)

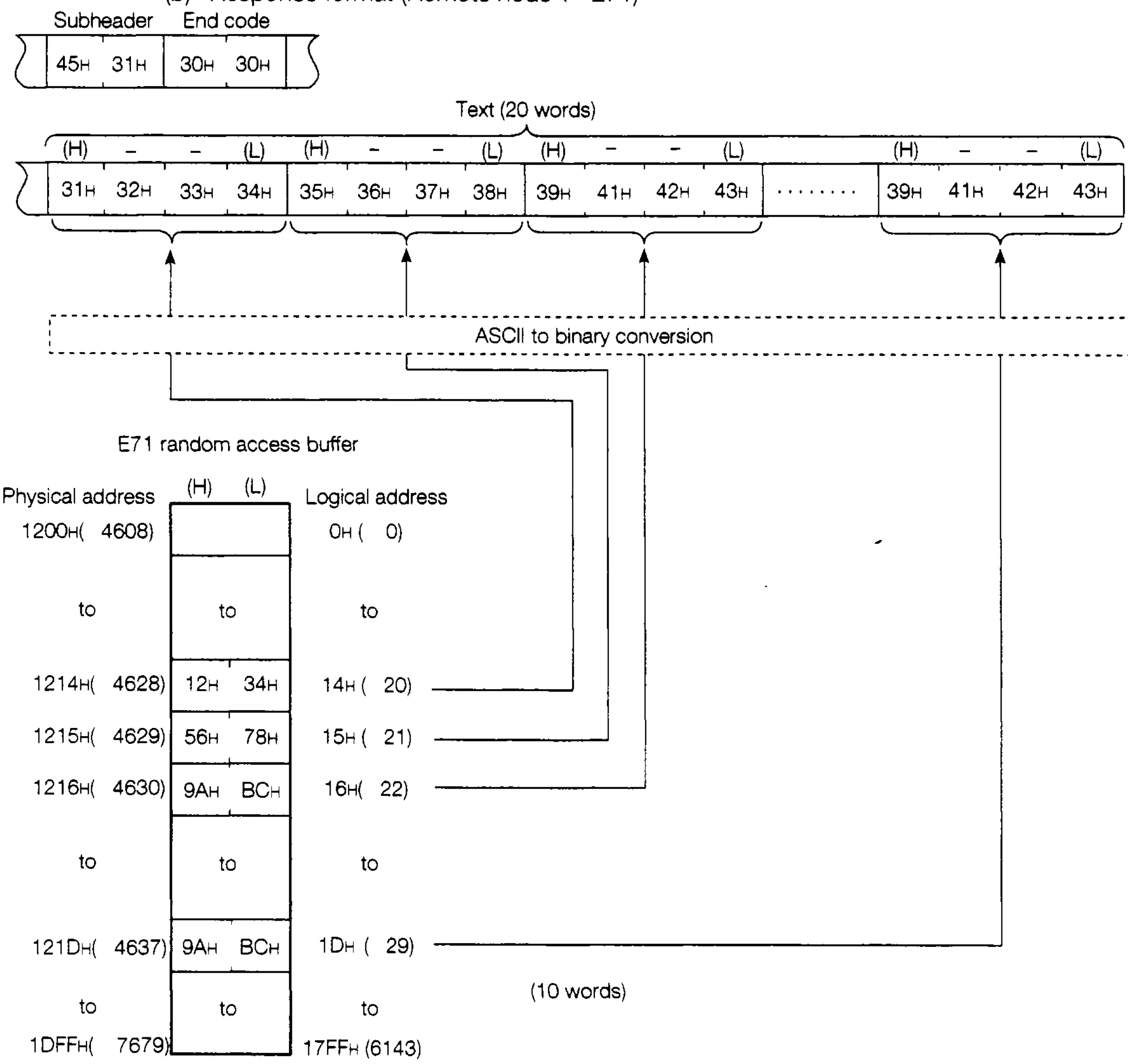


When ASCII code specified

(a) Command format (Remote node → E71)



(b) Response format (Remote node ← E71)



8.3 Programming

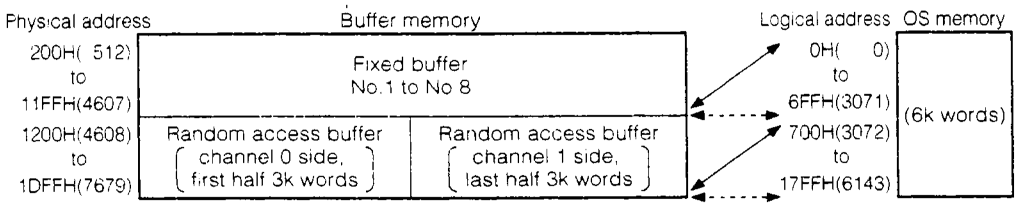
This section explains the programming for conducting exchange between the E71 and a remote node using the random access buffer.

8.3.1 Program Creation Precautions

(1) Exchange with a remote node using the random access buffer is conducted asynchronously with the PLC CPU program.

When synchronous exchange is required, conduct exchange by adding a free protocol between the partner remote node to which exchange will be conducted and the PLC CPU.

(2) For the random access buffer, the address specified by the remote node and the address specified by the sequence program's FROM/TO commands differ, so caution is required.



① Random access buffer address specified by the remote node

- The random access buffer can be read from or written to as a 6k word continuous area.
- The random access buffer address can be specified with the head area as 0000H (0) and the logical address as 0000H to 17FFH (0 to 6143).
- It is not necessary to recognize the channel switching signal (I/O signal: Y001C) on/off between the PLC CPU and the E71.

② Random access buffer address specified by the sequence program

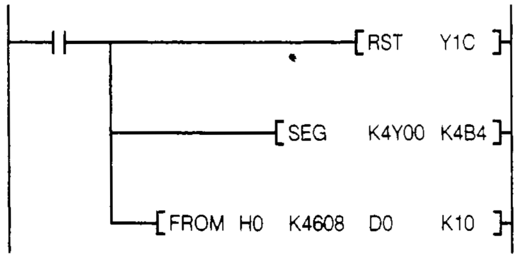
- The random access buffer can be read from/written to as a 3k byte area for each of the channel 0 and channel 1 side.
- The channel 0 side and channel 1 side random access buffers can be read from/written to after the channel switching signal (Y001C) on/off between the PLC CPU and the E71.

  Y001C = off : Read/write to the channel 0 side random access buffer.

  Y001C = on : Read/write to the channel 1 side random access buffer.
- Set the random access buffer address to physical address 1200H to 1DFFH (4608 to 7679).

- When the I/O control method of the PLC CPU of the station installed in the E71 is the refresh method and when random access buffer read/write is conducted after the buffer memory channel switching signal (Y001C) is changed from on → off/off → on, a read/write must be performed after outputting the channel switching signal (Y001C) to the E71. Output the channel switching signal (Y001C) to the E71 using the tail*1 of Item 3.6.2.

  (Example) When the common instruction's SEG is used.



(3) For command transmission, the next command should be sent after the completion of data communication (such as after the reception of a response) for the transmission of the previous command.

8.4.2 Program Creation Procedure

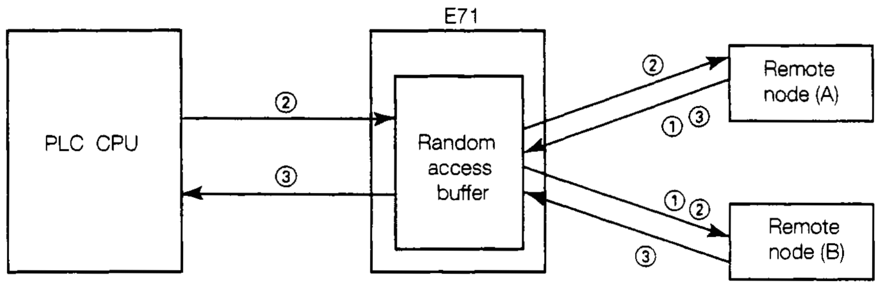
This section explains about the random access buffer exchange procedure.

As shown below, there are three exchange methods that can be used for random access buffer exchange.

① Remote node (A) or (B) reads the data written into the E71's random access buffer by remote node (A).

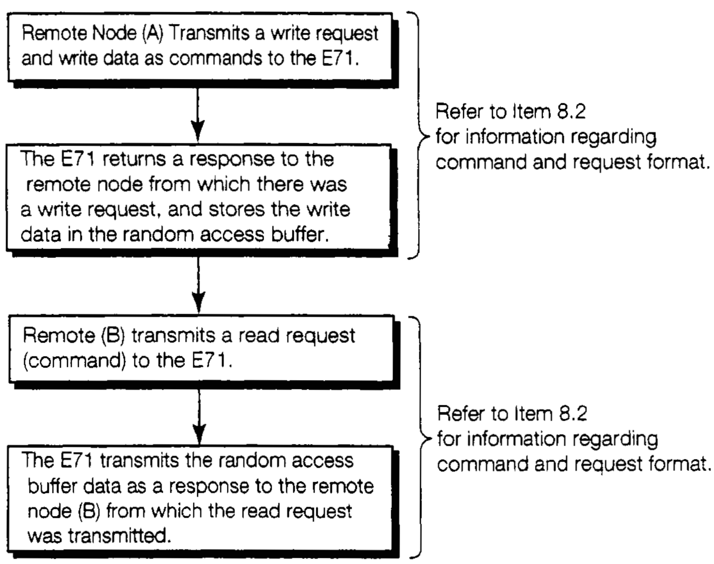
② Remote nodes (A) and (B) read the data written in the E71's random access buffer by the sequence program.

③ The sequence program reads the data written in the E71's random access buffer by remote node (A) or (B).



Following is an explanation of the exchange procedure for three exchange methods described above.

1 Exchange method where remote (B) reads the data written by remote node (A)



2 **Communication procedure when the remote node reads the data written by the PLC program**

The sequence program TO command writes data to the E71's random access buffer.

↓

A remote node transmits a read request (command) to the E71.

↓

The E71 transmits as a response the random access buffer data to the remotes node from which the read request was received.

Refer to Item 8.2 for information regarding command and response format.

3 **Exchange procedure when the data written by a remote node is read by the PLC program**

The remote node transmits a write request and write data as command to the E71

↓

The E71 returns a response to the remote node from which the write request was transmitted, and stores the write data in the random access buffer.

↓

The E71's random access buffer data is read by a sequence program FROM command.

Refer to Item 8.2 for information regarding commands and response format.

Remarks

With random access buffer exchange, a handshake cannot be conducted using the E71's I/O signal.

MEMO

READING/WRITING DATA IN THE PLC CPU SECTION

The reading and writing data in the PLC CPU section describes the method for reading and writing device memory and programs in the PLC CPU by the remote node's external devices via the Ethernet interface module, and the method for conducting remote control of the PLC CPU.

The read/write of data in the PLC CPU can be conducted regardless of the PLC CPU's RUN/STOP status when the data exchange function is used while the PLC CPU is stopped.

After connecting the communication line using the initial processing and open processing shown in Chapter 5 conduct a read/write of data in the PLC CPU.

In addition, conduct close processing and end processing when the data exchange ends in the corresponding communication line.

When conducting read/write of data in the PLC CPU, first read the explanation regarding common items in Chapter 9.

The read/write using E71 is explained in Chapter 10.

9. READING/WRITING DATA IN THE PLC CPU EXCHANGE

This section explains the control method, command list, and data exchange precautions when reading and writing PLC device and program data, etc., via the E71 from a remote node.

9.1 Control Method

This section explains the control method when reading and writing data in the PLC CPU.

1 **Reading and writing data in the PLC CPU can be performed regardless of the E71's I/O signal on/off state and the existence of the data exchange sequence program.**

2 **When writing to the PLC CPU from a remote node, the write approval/prohibition during PLC CPU RUN, can be selected using the CPU exchange timing setting switch on the front of the E71.**

Exchange timing setting switch (Refer to Item 4.3.2)

SW7/SW3 OFF : Writing from a remote node is not possible during PLC CPU RUN.

ON : Writing from the remote node is possible during both PLC CPU RUN/STOP.

3 **Data can be read from and written to the PLC CPU and special function unit by the remote node by transmitting the E71 commands described in Item 9.2 to the E71. In addition, it is also possible to read and write data to the remote station PLC CPU and special functions units on the MELSECNET.**

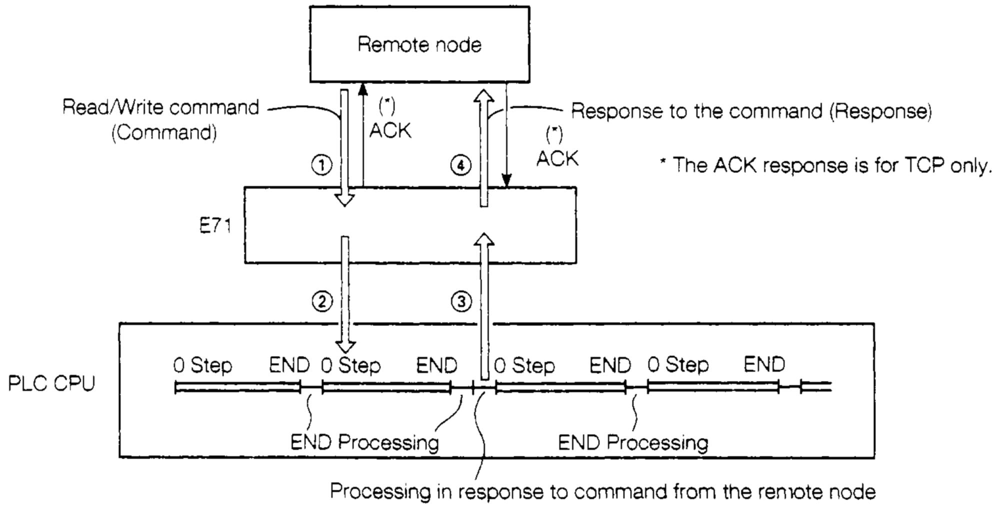
Point

(1) When writing to the special functions unit installed in the remote I/O stations in the data link system or network system from the remote node, the exchange timing setting switch (SW7/SW3) must be set to on. (The remote I/O station will change to the normal RUN state. You cannot switch between the RUN and STOP.)

(2) Refer to each system reference manual for details regarding the access possible range for remote station PLC on data link system or network system.

9.1.1 Exchanging with the PLC CPU Installed in the Ethernet Interface Module

1 **The control method for reading and writing data in the PLC CPU installed in the E71 is as follows.**



① The remote node transmits to the E71 a command (command) to read/write data in the PLC CPU.

② When the E71 receives the command from the remote node, it requests the read/write of data in the PLC CPU in accordance with the contents of the command.

③ When the sequence program's END command is executed, the PLC CPU follows the request from the E71 and conducts the data read/write and then transmits the processing results to the E71.

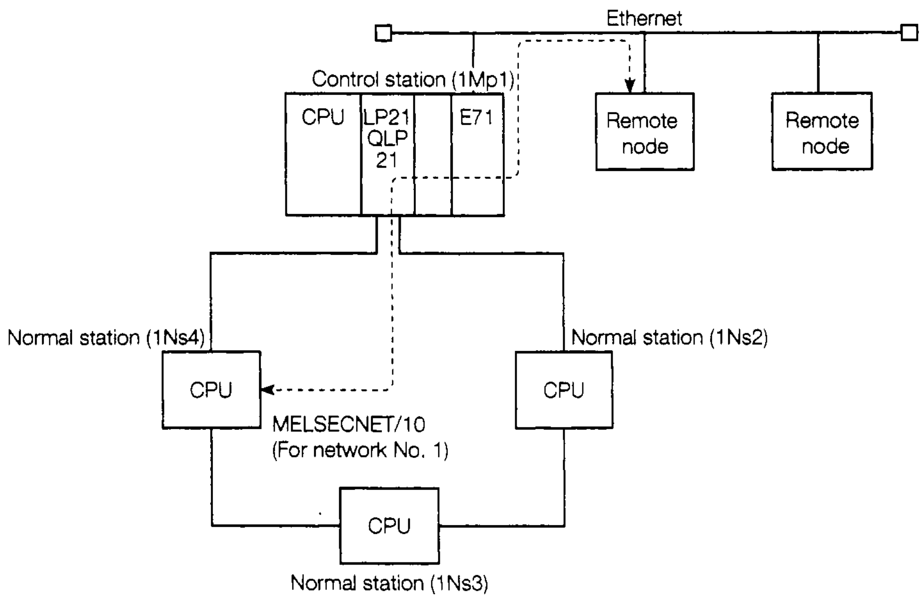
④ When the E71 receives the processing results from the PLC CPU, it sends a response that includes the processing results to the remote node from which the request originated.

Point

When read/write from the remote node is conducted during PLC CPU RUN, the processing time in response to the command from the remote node and the sequence program's scan time could become longer so caution is required.

9.1.2 Exchanging with the PLC CPU in the Network System

1 **When reading and writing data in the PLC CPU, reading and writing to the remote station PLC on the MELSECNET/10 can be done via the PLC CPU installed in the E71 within the network system's specification range.**



2 **The PLC that conducts read/write is specified in the PLC No. (FFH, 00H to 40H) in the command text.**

	Remote Node Access Station	PLC No. Specified by Remote Node
1	E71 installed station (Local station)	FFH
2	Network control station between PLC on the MELSECNET/10(Other station) (When the E71 is installed in the write normal station in the network between PLC)	0H (0)
	Remote I/O net's master station on the MELSECNET/10(Other station) (When the E71 is installed and remote I/O net remote station)	
3	Station on the MELSECNET/10 (Except for 1 and 2 above)	01H to 40H (1 to 64) (Access station No.)

Point

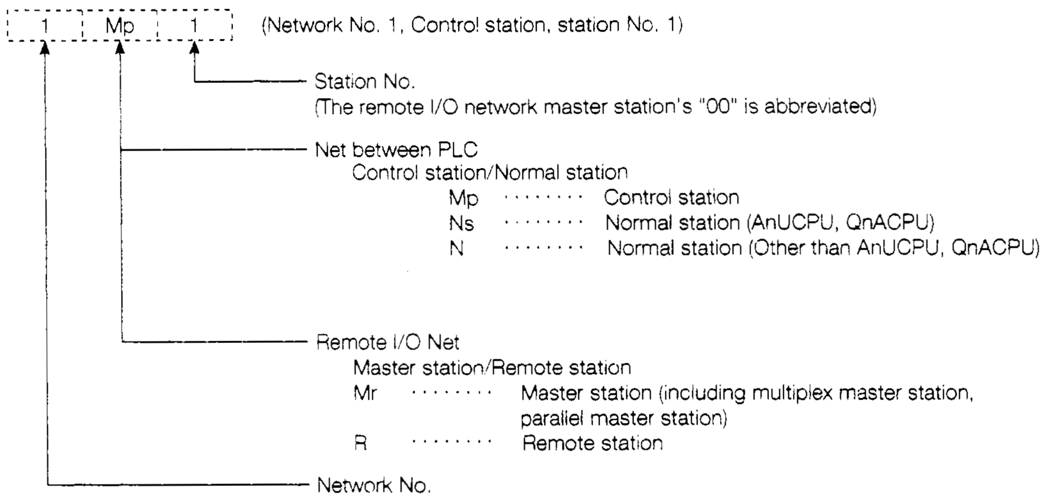
(1) When the station installed in the E71 is a base AnU/QnACPU and remote station access via the station installed in the E71 is conducted, the following parameters are set in the PLC CPU of the station installed in the E71 using the GPP function.
* "Valid unit during remote station access" setting : Set in the number of units setting, and set the unit through its exchange will pass during remote station access.

(2) When multiple network units are installed for the same network No. in the E71 installation station, remote station access is done via the network module installed in the base unit's slot of the newest No. when the network No. is specified.

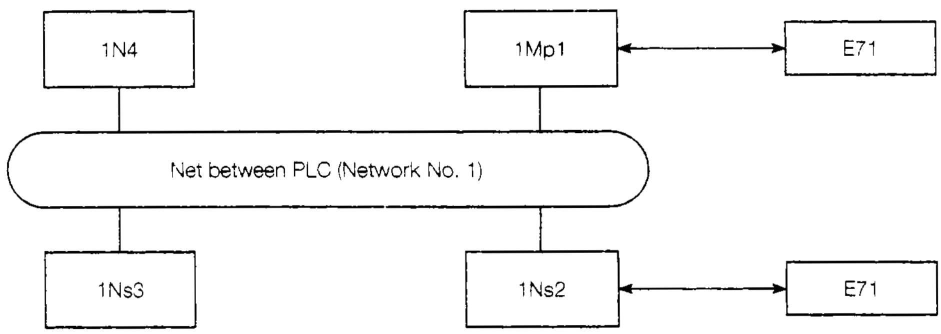
3 **Of the other stations in the network system those for which exchange with a PLC is possible are shown below. The exchange possible PLC vary depending on the stations installed with the E71.**

(Meaning of station symbols shown in the diagram)

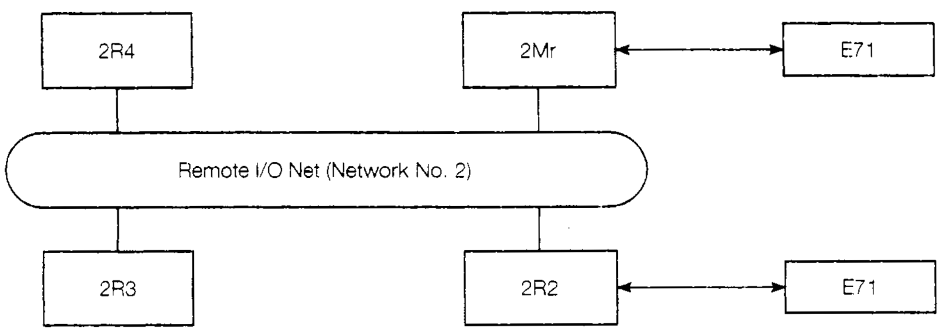
- Network system (MELSECNET/10)



(a) When the E71 installed station is the net between PLC command station/normal station, and when the remote I/O net master station



(b) When the E71 installed station is the remote I/O net's remote station



(c) PLC No. when E71 commands are used

PLC installed in the E71	Exchange possible PLC and PLC No. item specification values (Hexadecimal numbers)								
	Local Station	1Mp	1Ns2	1Ns3	1N4	2Mr	2R2	2R3	2R4
1Mp1	FF	—	02	03	04				
1Ns2	FF	01	—	03	04				
2Mr	FF					—	02[*1]	03[*1]	04[*1]
2R2	FF[*2]					00	—	×	

n .... Access to all devices is possible by setting the subject's PLC No.

n[*1] ..... Access to the special function unit buffer memory is possible by setting the subject PLC No.

n[*2] ..... The following devices can be accessed by setting the appropriate PLC No.

Device	Access possible device range	Remarks
Input	X0 to X7FF	—
Output	Y0 to Y7FF	
Link relay	B0 to BFFF	
Link register	W0 to WFFF	
Internal relay	M0 to M511	Device for switching SB0 to SB1FF
	M9000 to M9255	—
Data register	D0 to D511	Device for switching SW0 to SW1FF
	D9000 to D9255	—

x .... Access is not possible.

4 Transmission time when via network system

(a) The transmission time (T1) when data is transmitted to a PLC on a network system in which an E71 is not installed is shown below.

① For net between PLC

- Transmission time (T1) = ($\underbrace{\text{Transmission delay time}}_{*1}$ + E71 installed station 1 scan time) × ($\underset{*2}{n}$ + $\underset{*3}{1}$)

*1 Refer to the network system reference manual for an explanation of the transmission delay time.

*2
- When initial exchange is conducted for the subject station when the power is turned on and after the CPU is reset.
- When exchange is conducted to the station except the 10 stations most recently exchange with.

} n = 6

- When exchange is conducted the second time when the number of exchange stations is under 10.
- When exchange is conducted the second time to the first 10 stations.

} n = 1

*3 When the "CPU communication timing setting" of the E71 communication condition setting switch is off (write prohibited), it is added only when data is written from remote node.

- Reason for transmission time (T1) delay

When commands that require two scans (device write, etc when the DIP switch SW7/SW3 is off, etc.) are executed, then the value is calculated using the above formula. Refer to Item 9.3 2 for the required number of scans when a remote E71, GPP function, etc. requests access to the same PLC CPU at the same time.

- Increase the CPU monitoring timer's monitoring time from the other station when exchange to other station is conducted via MELSECNET/10.

- For details regarding network systems, refer to the network system reference manuals.

(Example) When an E71 is installed in the station on the MELSECNET/10 (net between PLC) and the device memory for a remote system on the same MELSECNET/10 is read. (Second exchange time when the number of exchange stations is under 10)

- ST : Transmission scan time 120ms
- α T : Transmission link refresh time 10ms
- SR : Receive scan time 100ms
- α R : Receive link refresh time 5ms
- LS : Link scan 30ms
- Number of simultaneous transient requests : 3
- Maximum number of times for transient : 2 (Using the user set value when for the QnACPU.)

$$\text{Transmission Time (T1)} = \{\underset{(ST)}{120} \times 2 + \underset{(\alpha T)}{10} \times 2 + \underset{(LS)}{30} \times 6 + \underset{(SR)}{100} \times 2 + \underset{(\alpha R)}{5} \times 2$$

$$+ \left(\underbrace{\frac{3\ \text{(Number of simultaneous transient requests)}}{2\ \text{(Maximum number of transient times)}}}_{\text{Adjustment value (Decimal round off)}} -1 \right) \times \underset{(LS)}{30} \times 2 + \underset{(ST)}{120}\} \times 1 = 890\text{ms}$$

② For Remote I/O Net

- Transmission Time (T1) = $\underbrace{\text{(Transmission delay time}}_{*1}$ + 1 link scan time) × ($\underbrace{n}_{*2}$ + $\underbrace{1}_{*3}$)

*1 Please refer to the explanation of transmission delay time in the network system reference manual.

*2
- When exchange is conducted for the first time to the subject station after link is begun.
- When conducting exchange to a station except the latest 10 stations to which exchange was conducted.

} n = 6

- When conducting exchange for the second time when the number of exchange stations is under 10.
- When conducting exchange for the second time to the latest 10 stations in which exchange was conducted.

} n = 1

*3 When the "CPU communication timing setting" of the E71 communication condition setting switch is off (write prohibited), it is added only when data is written from remote node.

- Reason For Transmission Time (T1) Delay

  When commands that require two scans (device write, etc. when the DIP SW7/SW3 is off) are transmitted, the value is twice that calculated using the above formula.

  Refer to Item 9.3 [2] for information regarding the necessary number of scans when request access to the same PLC CPU is made at the same time by a remote E71, GPP function, etc.

- Increase the CPU monitoring timer's monitoring time from the other station when exchange to other station is conducted via MELSECNET/10.

  * Refer to the network system reference manual for details regarding network system.

(Example) When an E71 is installed in a station on an MELSECNET/10 and read from the other station device memory is conducted on the same MELSECNET/10.

(Second exchange time when the number of exchange stations is under 16)

- Sm : Master station sequence scan time 120ms
- α m : Master station link refresh time 10ms
- α r : Remote I/O station link refresh time 2ms
- LS : Link scan time 30ms

Because the above (Sm) > (LS) the formula is as follows. (When there is one master station)

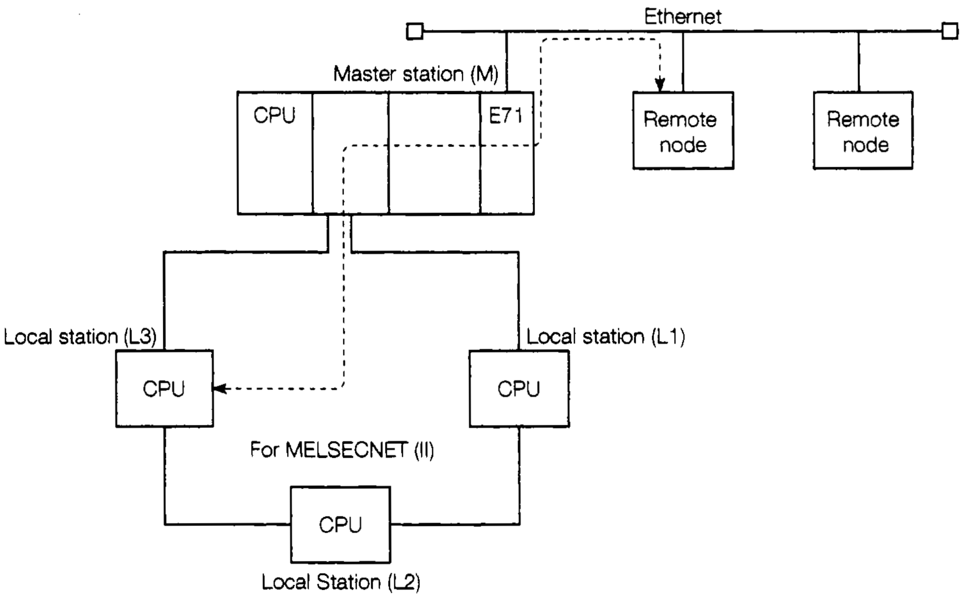
Transmission Time (T1) = { ($\underset{(Sm)}{120}$ + $\underset{(\alpha\ m)}{10}$) × 3 + $\underset{(LS)}{30}$ } × 1 = 420ms

Point

There will be an appropriate delay corresponding to the conditions during data transmission to a PLC in which an E71 is not installed on the MELSECNET/10.
The transmission delay time for exchange with the PLC can be reduced by using the E71 installed station (PLC No.FF_H) only and using the MELSECNET/10 data link (LB, LW) for exchange with remote station PLC.

9.1.3 Exchanging with the PLC CPU in the Data Link System

1 **When reading and writing in the PLC CPU, reading and writing can be done to the other station PLC in the MELSECNET(II) and MELSECNET/B via the PLC CPU in which an E71 is installed within the data link system specification range.**



2 **The PLC that conducts read/write is specified in the PLC No. (FFH, 00H to 40H) in the command text.**

	Remote Node Access Station	PLC No. Specified by the Remote Node
1	When E71 is installed (Local station)	FFH
2	Master station on the MELSECNET(II) (Other station) (Except 1 above)	00H
3	Local station/Remote station on the MELSECNET(II) (Other station) (Except 1 and 2 above)	01H to 40H (1 to 64) (Access station's station No.)
4	Other station on the MELSECNET/B	(Same as 2 and 3 above)

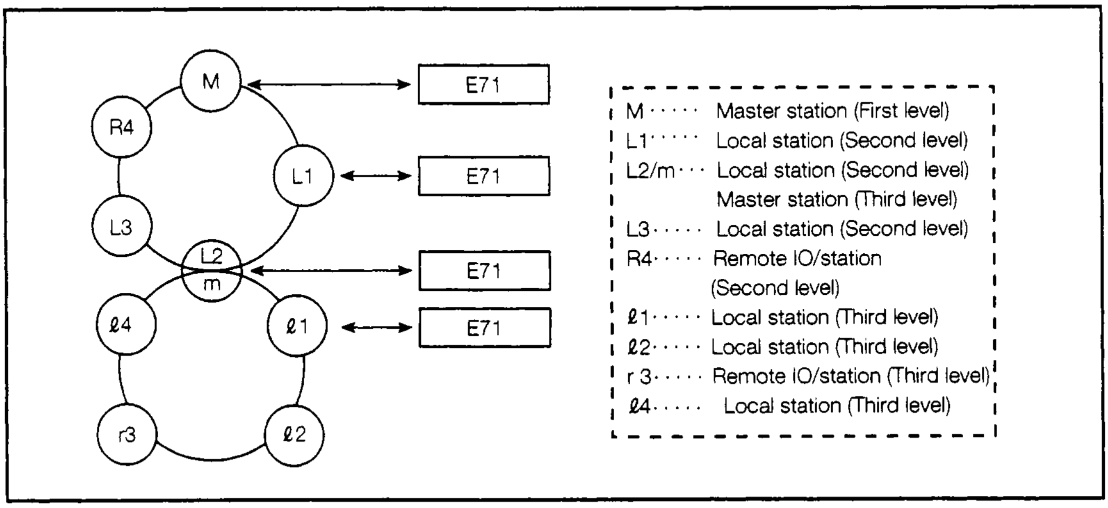
Point

(1) When the station installed in the E71 is a base AnU/QnACPU and other station access via the station installed in the E71 is conducted, the following parameters are set in the PLC CPU of the station installed in the E71 using the GPP function.

- "Valid units during remote station access" setting
  .... Sets the unit value which exchange will be conducted for other station access in the settings and the number of unit settings.

3 **The following shows the exchange possible PLC of the other stations in the data link system.**

The exchange possible stations vary according to the stations with E71 installed.



PLC No. when using E71 commands

PLC Installed in the E71	Exchange Possible PLC and Subject Item Specification Values (Hexadecimal Numbers)									
	Local Station	**M**	**L1**	**L2/m**	**L3**	**R4**	**ℓ1**	**ℓ2**	**r3**	**ℓ4**
M	FF	—	01	02	03	04*1		×		
L1	FF	00	—				×			
L2/m	FF	00	×	—	×		01	02	03*1	04
ℓ1	FF	×		00	×		—		×	

n ..... All devices can be accessed by specifying the subject PLC's No.

n*1 ... The special function unit' buffer memory can be accessed by specifying the subject PLC's No.

x ..... Access not possible

Point

Exchange cannot be done with A0J2CPUP23/R23 and A0J2P25/R25.

4 **Transmission time when done via data link system**

(a) The following shows the transmission time (T1) when data transmission is conducted to a PLC in which an E71 is not installed in the data link system.

- Local station

  Transmission Time (T1) = (Transmission delay time A + E71 installed 1 station scan time) × (n+1)
  *1 *2 *3

- Remote I/O station

  Transmission time (T1) = (Transmission delay time B + Master station 1 scan time) × (n+1)
  *1 *2 *3

*1 Refer to the explanation on the subject data link system transmission delay time in the data link system reference manual.

Transmission delay time A : ...... Refer to the symbol column for the LRDP command processing time

Transmission delay time B : ...... Refer to the symbol column for the RFRP command processing time

*2
- When exchange is conducted for the first time to the subject system when the power is turned on or when the CPU reset has been conducted.
- When exchange is conducted with a station other than the latest 10 stations to which exchange has been conducted.

n = 3

- When exchange is conducted for the second time when the number of exchange stations is under 10.
- When exchange is conducted for the second time to the latest 10 stations to which exchange has been conducted.

n = 1

*3 When the "CPU communication timing setting" of the E71 communication condition setting switch is off (write prohibited), it is added only when data is written from remote node.

- Reason for transmission time (T1) delay

  When commands that require two scans (device write, etc. when the DIP switch SW7/SW3 is off, etc.) are executed, then the value is calcuated using the above formula.

  Refer to Item 9.3 2 for the required number of scans when a remote E71, GPP function, etc.

- Lengthen the CPU monitoring timer's monitoring time from the other station when exchange is conducted with the other station via the data link system.

  - Refer to the data link system reference manual for details regarding data links.

(Example) When the E71 is installed in the MELSECNET (II) master station, and the local station's device memory is read.

(Conditions $L < LS < M$, M : 80ms $\alpha1$: 10ms)

Transmission Time (T1) = $(M \times 4 + \alpha1 \times 4 + M) \times 1 = (80 \times 4 + 10 \times 4 + 80) \times 1 = 440$

Tl is 880ms.

M : MELSECNET master station scan time

$\alpha1$: MELSECNET master station link refresh time

LS : Link scanner time

L : MELSECNET local station's scan time

Point

Depending on conditions, a considerable delay can occur in data transmission to the PLC in which an E71 is not installed on the MELSECNET.
The transmission delay time can be reduced by using only the E71 installed station (PLC No. FF_H) for exchange with the PLC, and by using a data link (B, W) for exchange with the other station PLC CPU.

9.1.4 Exchanging with the PLC CPU in Mixed Systems

Reading and writing cannot be done to the following other station PLC.

① Other station PLC on data link systems via network systems.

② Other station PLC on network systems via data link systems.

9.2 List of E71 Commands and Functions

This section explains the commands and functions used to read and write data in the PLC CPU from remote nodes.

Functions			Command Response Types	Description of Processing	Number of Processes Performed For 1 Exchange
Device memory *4	Batch read	Bit unit	00H	Bit devices (X, Y, M, etc.) are read in 1 point unit.	256 Points
		Word unit	01H	Bit devices (X, Y, M, etc.) are read in 16 point units.	128 Words (2048 Points)
				Word devices (D, R, T, C, etc.) are written in 1 point unit. *3	256 Points
	Batch write	Bit unit	02H	Bit devices (X, Y, M, etc.) are written in 1 point unit.	256 Points
		Word unit	03H	Bit devices (X, Y, M, etc.) are written in 16 point units.	40 Words (640 Points)
				Word devices (D, R, T, C, etc.) are written in 1 point unit. *3	256 Points
	Test (Random write)	Bit unit	04H	Bit devices (X, Y, M, etc.) are set and reset in 1 point unit and the devices and device No. are randomly set.	80 Points
		Word unit	05H	Bit devices (X, Y, M, etc.) are set and reset in 16 point units, and the devices and device No. are randomly set.	40 Words (640 Points)
				Word devices (D, R, T, C, etc.) are written in 1 point units, and the devices and device No. are randomly set.	40 Points
	Monitor data registration	Bit unit	06H	Bit devices that monitor (X, Y, M, etc.) are registered in 1 point unit.	40 Points *2
		Word unit	07H	Bit devices that monitor (X, Y, M, etc.) are registered in 16 point units.	20 Words (320 Points) *2
				Word devices that monitor (D, R, T, C, etc.) are registered in 1 point unit.	20 Points
	Monitor	Bit unit	08H	Device monitors for which monitor data registration was conducted.	(Number of registrations portion)
		Word unit	09H		
Extension file register	Batch read		17H	Extension file registers (R) are read in 1 point unit.	256 Points
	Batch write		18H	Extension file registers (R) are written in 1 point unit.	256 Points
	Test (Random write)		19H	Extension file registers (R) are written in 1 point unit and the block No. and device No. are randomly set.	40 Points
	Monitor data registration		1AH	The extension file registers that monitor (R) are registered in 1 point unit.	20 Points
	Monitor		1BH	Monitors the extension file registers (R) that conduct the monitor data registration.	—
	Direct read		3BH	Reads in 1 point unit the extension file registers (R) that are directly set.	256 Points
	Direct write		3CH	Reads in 1 point unit the extension file registers (R) that are directly set.	256 Points
Special function Module	Batch read		0EH	Reads the contents of the special function module buffer memory.	256 Bytes(128 Words)
	Batch write		0FH	Writes the data to the special function module buffer memory.	256 Bytes(128 Words)

Subject PLC CPUs that can Execute Commands																		PLC CPU State [*1]			Reference Item
A0J2	A0J2H	A1, A1N	A2, A2N A2S (S1)	A3, A3N A1SH A1SJH A2SH (S1)	A3H, A3M	A2C, A2CJ	A1S, (S1) A1SJ	A2A (S1)	A3A	A2U, A2AS (S1)	A3U	A4U	AJ72 P25/ R25	QLP 25 LP25 / QBR 15 BR15	Q2A Q2AS Q2ASH (S1)	Q3A	Q4A Q4AR	During STOP	During RUN Write Possible Setting	During RUN Write not Possible Setting	
×	○												×	○	○			○	○	○	Item 10.2.2
×	○												×	○	○						Item 10.2.3
×	○												×	○	○						
×	○												×	○	○			○	○	×	Item 10.2.4
×	○												×	○	○						Item 10.2.5
×	○												×	○	○						
×	○												×	○	○			○	○	×	Item 10.2.6
×	○												×	○	○						Item 10.2.7
×	○												×	○	○						
×	○												×	○	○			○	○	○	Item 10.2.8
×	○												×	○	○						
×	○												×	○	○						
×	○												×	○	○			○	○	○	
×	○	×	○										×					○	○	○	Item 10.3.3
×	○	×	○										×					○	○	×	Item 10.3.4
×	○	×	○										×					○	○	×	Item 10.3.5
×	○	×	○										×					○	○	○	Item 10.3.6
×	○	×	○										×								
×	○	×	○										×								Item 10.3.7
×	○	×	○										×					○	○	×	
×															×			○	○	○	Item 10.4.2
×															×			○	○	×	Item 10.4.3

Functions				Command Response Types	Description of Processing	Number of Processes Performed For 1 Exchange
Sequence program *6	Batch read	Main	Sequence program	0AH	Reads the main sequence program.	256 Steps
			T/C set value		Reads the T/C set value used by the main sequence program.	256 Points
		Sub	Sequence program	0BH	Reads the sub sequence program.	256 Steps
			T/C set value		Reads the T/C set value used by the sub sequence program.	256 Points
	Batch write	Main	Sequence program	0CH	Writes the main sequence program.	256 Steps
			T/C set value		Writes the T/C set value used by the main sequence program.	256 Points
		Sub	Sequence program	0DH	Writes the sub sequence program.	256 Steps
			T/C set value		Writes the T/C set value used by the sub sequence program.	256 Points
Microcomputer program	Batch read		Main	1EH	Reads the main microcomputer program.	256 Bytes
			Sub	1FH	Reads the sub microcomputer program.	
	Batch write		Main	20H	Writes the main microcomputer program.	
			Sub	21H	Writes the sub microcomputer program.	
Comment	Batch read			1CH	Reads the comment data.	256 Bytes
	Batch write			1DH	Writes the commend data.	
Extension comment	Direct read			39H	Reads the extension comment data.	256 Bytes
	Direct write			3AH	Writes the extension comment data.	
Parameter *7	Batch read			10H	Reads the PLC CPU parameter contents.	256 Bytes
	Batch write			11H	Writes the parameter contents to the PLC CPU.	
	Analysis request			12H	Recognizes and checks the overwritten parameter contents in the PLC CPU.	—
PLC CPU	Remote RUN			13H	Requests a remote RUN/STOP of the PLC CPU.	—
	Remote STOP			14H		
	PLC model read			15H	Reads whether the PLC CPU model is A1N, A2N, A3N, A3H... .	
Loopback test				16H	The characters received from the remote node are returned to the remote node.	256 Bytes

<table>
<tr><th colspan="18">Subject PLC CPUs that can Execute Commands</th><th colspan="3">PLC CPU State *1</th><th rowspan="3">Reference Item</th></tr>
<tr><th rowspan="2">A0J2</th><th rowspan="2">A0J2H</th><th rowspan="2">A1, A1N</th><th rowspan="2">A2, A2N A2S (S1)</th><th rowspan="2">A3, A3N A1SH A1SJH A2SH (S1)</th><th rowspan="2">A3H, A3M</th><th rowspan="2">A2C, A2CJ</th><th rowspan="2">A1S, (S1) A1SJ</th><th rowspan="2">A2A (S1)</th><th rowspan="2">A3A</th><th rowspan="2">A2U, A2AS (S1)</th><th rowspan="2">A3U</th><th rowspan="2">A4U</th><th rowspan="2">AJ72 P25/ R25</th><th rowspan="2">QLP 25 LP25 / QBR 15 BR15</th><th rowspan="2">Q2A Q2AS Q2ASH (S1)</th><th rowspan="2">Q3A</th><th rowspan="2">Q4A Q4AR</th><th rowspan="2">During STOP</th><th colspan="2">During RUN</th></tr>
<tr><th>Write Possible Setting</th><th>Write not Possible Setting</th></tr>
<tr><td>×</td><td colspan="12">○</td><td colspan="2">×</td><td colspan="3">×</td><td rowspan="2">○</td><td rowspan="2">○</td><td rowspan="2">○</td><td rowspan="8">Item 10.6.4</td></tr>
<tr><td>×</td><td colspan="12">○</td><td colspan="2">×</td><td colspan="3">×</td></tr>
<tr><td>×</td><td colspan="2">×</td><td colspan="2">○</td><td colspan="3">×</td><td>○</td><td>×</td><td colspan="2">○</td><td colspan="2">×</td><td colspan="3">×</td><td rowspan="2">○</td><td rowspan="2">○</td><td rowspan="2">○</td></tr>
<tr><td>×</td><td colspan="2">×</td><td colspan="2">○</td><td colspan="3">×</td><td>○</td><td>×</td><td colspan="2">○</td><td colspan="2">×</td><td colspan="3">×</td></tr>
<tr><td>×</td><td colspan="10">○</td><td colspan="2">×</td><td colspan="2">×</td><td colspan="3"></td><td>○</td><td>○ *5</td><td>×</td></tr>
<tr><td>×</td><td colspan="10">○</td><td colspan="2">×</td><td colspan="2">×</td><td colspan="3"></td><td>○</td><td>○</td><td>×</td></tr>
<tr><td>×</td><td colspan="2">×</td><td colspan="2">○</td><td colspan="3">×</td><td>○</td><td>×</td><td colspan="2">○</td><td colspan="2">×</td><td colspan="3">×</td><td>○</td><td>○ *5</td><td>×</td></tr>
<tr><td>×</td><td colspan="2">×</td><td colspan="2">○</td><td colspan="3">×</td><td>○</td><td>×</td><td colspan="2">○</td><td colspan="2">×</td><td colspan="3">×</td><td>○</td><td>○</td><td>×</td></tr>
<tr><td>×</td><td colspan="10">○</td><td colspan="2">×</td><td colspan="2">×</td><td colspan="3"></td><td rowspan="2">○</td><td rowspan="2">○</td><td rowspan="2">○</td><td rowspan="4">Item 10.6.5</td></tr>
<tr><td>×</td><td colspan="2">×</td><td colspan="2">○</td><td colspan="8">×</td><td colspan="2">×</td><td colspan="3">×</td></tr>
<tr><td>×</td><td colspan="10">○</td><td colspan="2">×</td><td colspan="2">×</td><td colspan="3"></td><td rowspan="2">○</td><td rowspan="2">○ *5</td><td rowspan="2">×</td></tr>
<tr><td>×</td><td colspan="2">×</td><td colspan="2">○</td><td colspan="8">×</td><td colspan="2">×</td><td colspan="3">×</td></tr>
<tr><td>×</td><td colspan="10">○</td><td colspan="2">×</td><td colspan="2">×</td><td colspan="3"></td><td>○</td><td>○</td><td>○</td><td rowspan="2">Item 10.6.6</td></tr>
<tr><td>×</td><td colspan="10">○</td><td colspan="2">×</td><td colspan="2">×</td><td colspan="3"></td><td>○</td><td>○</td><td>×</td></tr>
<tr><td>×</td><td colspan="7">×</td><td colspan="3">○</td><td colspan="2">×</td><td colspan="2">×</td><td colspan="3"></td><td>○</td><td>○</td><td>○</td><td rowspan="2">Item 10.6.7</td></tr>
<tr><td>×</td><td colspan="7">×</td><td colspan="5">○</td><td colspan="2">×</td><td colspan="3">×</td><td>○</td><td>○</td><td>×</td></tr>
<tr><td>×</td><td colspan="12">○</td><td colspan="2">×</td><td colspan="3">×</td><td>○</td><td>○</td><td>○</td><td rowspan="3">Item 10.6.3</td></tr>
<tr><td>×</td><td colspan="12">○</td><td colspan="2">×</td><td colspan="3">×</td><td>○</td><td>×</td><td>×</td></tr>
<tr><td>×</td><td colspan="12">○</td><td colspan="2">×</td><td colspan="3">×</td><td>○</td><td>×</td><td>×</td></tr>
<tr><td>×</td><td colspan="12">○</td><td colspan="2">×</td><td colspan="3">○</td><td>○</td><td>○</td><td>○</td><td>Item 10.5.2</td></tr>
<tr><td>×</td><td colspan="14">○</td><td colspan="3">○</td><td>○</td><td>○</td><td>○</td><td>Item 10.5.3</td></tr>
<tr><td>—</td><td colspan="14">—</td><td colspan="3">—</td><td>○</td><td>○</td><td>○</td><td>Item 10.7</td></tr>
</table>

*1 Use the E71's DIP switch SW07/SW03 to set whether it is possible to write to the PLC CPU during RUN.

SW07/SW03 = ON .... Write possible during RUN (Possible)
SW07/SW03 = OFF .... Write not possible during RUN (Forbidden)

*2 For other than A3HCPU, AnA, AnU, QnACPU, 2 points worth of points are processed for each point for device X (input).

When X is included in the set device, make it as follows

((Number of specified points × 2) + number of other device set points) ≤ number of points processed per one exchange

When only X is specified, the number of points that can be processed per exchange is one half the value shown in the table.

*3 When reading or writing extension file registers, use the extension file register's special command.

*4 The AnUCPU can be accessed using the AnACPU device range. Only devices with the same name as the devices existing in the AnACPU can be accessed in the QnACPU using the AnACPU device range. (Except below)

The following QnACPU devices cannot be accessed from remote node:

- Devices newly added to the QnACPU
- Latch relay (L) and step relay (S)
  - For the QnACPU, the latch relay (L) and step relay (S) are separete devices from internal relays (M), but access will be made to internal relays when either one is specified.
- File register (R)

QnACPU Accessible Devices (Accessible with E71 Commands)....When the Parameter Settings are the Default

Classification	Device	Device No. (Settings range)	Decimal/ Hexadecimal Expression	Classification	Device		Device No. (Settings range)	Decimal/ Hexadecimal Expression
Internal user device	Input relay	X0 to X7FF	Hexadecimal expression	Internal user device	Timer	Contact point	TS0 to TS2047	Decimal expression
	Output relay	Y0 to Y7FF				Coil	TC0 to TC2047	
	Internal relay	M0 to M8191	Decimal expression			Current value	TN0 to TN2047	
	Link relay	B0 to BFFF	Hexadecimal expression		Counter	Contact point	CS0 to CS1023	
	Enunciator	F0 to F2047	Decimal expression			Coil	CC0 to CC1023	
	Data register	D0 to D6143				Current value	CN0 to CN1023	
	Link register	W0 to WFFF	Hexadecimal expression	Internal system device	Special relay **1		M9000 to M9255	
					Special register **2		D9000 to D9255	

**1 Access for SM1000 to SM1255 is set at M9000 to M9255.

**2 Access for SD1000 to SD1255 is set at D9000 to D9255.

*5 Conduct program write during RUN when all of the following conditions are met.

① The PLC CPU is A3, A3N, A3A, A3U, or A4U.

② The program is not a program that is running. (Shows a subprogram when the main program is running.)

③ The PLC CPU special relay is in the following states.

(a) M9050 (signal flow exchange point) .... OFF (A3CPU only)

(b) M9051 (CHG instruction execution prohibited) .... ON

*6 Read/write cannot be conducted for the A4UCPU subprogram's sub 2 to sub 4.

*7 The parameter capacity for AnUCPU is 3k byte + MELSECNET/10 parameter (max. 24k byte).

9.3 PLC CPU Operation during Data Exchange

This section explains the PLC CPU operation when reading and writing data to the PLC CPU is conducted.

1 **PLC CPU scan time**

Access to the E71 and PLC CPU is processed once for each request for each END processing when the PLC CPU is running when a request is received from the E71. Therefore, this will increase the scan time processing time. For information regarding the PLC CPU interrupt time required for communication between the E71 and the PLC CPU, refer to Appendix 3.

2 **Simultaneous access to the PLC CPU**

Only one request is processed for END processing by the PLC CPU.
When the same QnACPU is accessed at the same time from a unit and the GPP function, the access is made to wait until other processing is completed, so the number of scans required for the processing is increased. Placing COM commands in the sequence program will increase the COM command's execution time scan time making it possible to process multiple accesses within one scan.

Remarks

When set to the "PLC System Setting" of the QnACPU parameter of the station installed in the E71

If the "PS system settings" general data processing setting is conducted, the QnACPU will process the requests for general data processing settings using END processing.

(Example)

If the "PLC system setting" general data processing setting is "4," the QnACPU can process all of a maximum of 4 access requests from units and GPP functions during the scan's END processing time. In addition, putting in COM commands increases the COM command execution time's scan time, making it possible for the QnACPU to process all of a maximum of 4 access requests from units and GPP functions during the COM command's execution time.

9.4 Data Exchange Precautions

Following is a list of precaution items for when reading and writing data to the PLC CPU is conducted.

1 **Conduct read/write when the E71's initial normal end signal (X19) and the open end signal (X10 to X17) of the connection to be used are turned on. If these signals are on, then it is possible to conduct read/write of data in the PLC CPU from the remote node regardless of whether the PLC program is valid.**

2 **When writing data when the PLC CPU is running, set the CPU exchange timing setting switch (dip switch SW7/SW3) on the front of the E71 to on.**

3 **When conducting a PLC CPU remote stop, use the data exchange function (Refer to Item 5.6) while the PLC CPU is stopped.**

4 **When the usage availability of the connection being opened is without procedure, reading and writing data to the PLC CPU cannot be conducted.**

5 **Changing the remote station PLC CPU to which data will be exchanged.**

After the E71 is booted up, remote station PLC CPU information is read in and stored. To change the remote station PLC CPU to which data will be exchanged after the E71 is booted up, reboot the E71 after changing the PLC CPU's model name. (Local station PLC power reset/CPU reset)

6 **When transmitting a command for reading/writing data in the PLC CPU, send the next command after the completion of the data communication for the transmission of the previous command.**

7 **Do not change data, program or remote control's RUN and STOP while operating with a PC connected to the special function module.**
Please make sure that you have read this manual carefully or you may cause erroneous operation or failure.

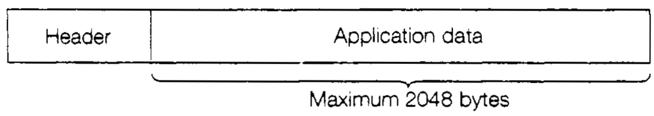
10. WHEN CONDUCTING READ/WRITE OF DATA IN THE PLC CPU

This section explains the control method and data format used to conduct reading/writing from the remote node via the E71 for devices and program data in the PLC CPU.

10.1 Data Format

Following is shown the exchange data (command and response) data item order and contents when conducting exchange by reading/writing data in the PLC CPU between the E71 and a remote node.

As is shown below, the exchange data consists of a header and application data.



10.1.1 Format When Exchanging Using Binary Code

Following shows the command and response data item order when exchange binary code data for the application data portion of the exchange data when exchanging by reading/writing data in the PLC CPU.

1 Transmission/reception data order when exchanging using TCP/IP

(a) Order during command transmission

Header			Application data (*1)			
Ethernet	IP	TCP	Subheader	PLC No.	ACPU monitor timer (L) (H)	Text (command)
(14 bytes)	(20 bytes)	(20 bytes)	(1 byte)	(1 byte)	(2 bytes)	(Maximum 2044 bytes)

Differs depending on the function (Application data)

(b) Order during response reception

Header			Application data (*1 *2)		
Ethernet	IP	TCP	Subheader	End code	Text (response)
(14 bytes)	(20 bytes)	(20 bytes)	(1 byte)	(1 byte)	

Differs depending on the function (Application data)

2 Transmission/reception data order when exchanging using UDP/IP

(a) Order during command transmission

Header			Application data (*1)			
Ethernet	IP	UDP	Subheader	PLC No.	ACPU monitor timer (L) (H)	Text (command)
(14 bytes)	(20 bytes)	(8 bytes)	(1 byte)	(1 byte)	(2 bytes)	(Maximum 2044 bytes)

Differs depending on the function (Application data)

(b) Order during response reception

Header			Application data (*1 *2)		
Ethernet	IP	UDP	Subheader	End code	Text (response)
(14 bytes)	(20 bytes)	(8 bytes)	(1 byte)	(1 byte)	

Differs depending on the function (Application data)

*1 The data order for each function and the data order when the status is normal are shown in each function explanation item from Item 10.2 and later.

*2 The application data portion data order is as follows when the response end code is "$5B_H$" (fault end).

Header	Application data			
	Subheader	End code $5B_H$	Error code	00_H
	(1 byte)	(1 byte)	(1 byte)	(1 byte)

10.1.2 Format When Exchanging Using ASCII Code

Following shows the command and response data item order when exchange ASCII code data for the application data portion of the exchange data when exchanging by reading/writing data in the PLC CPU.

1 Transmission/reception data order when exchanging using TCP/IP

(a) Order during command transmission

Header			Application data (*1)			
Ethernet	IP	TCP	Subheader (H) (L)	PLC No. (H) (L)	ACPU monitor timer (H) (L)	Text (command)
(14 bytes)	(20 bytes)	(20 bytes)	(2 bytes)	(2 bytes)	(4 bytes)	(Maximum 2040 bytes)

Differs depending on the function (application data portion)

(b) Order during response reception

Header			Application data (*1 *2)		
Ethernet	IP	TCP	Subheader (H) (L)	End code (H) (L)	Text (response)
(14 bytes)	(20 bytes)	(20 bytes)	(2 bytes)	(2 bytes)	

Differs depending on the function (application data portion)

2 Transmission/reception data order when exchanging using UDP/IP

(a) Order during command transmission

Header			Application data (*1)			
Ethernet	IP	UDP	Subheader (H) (L)	PLC No. (H) (L)	ACPU monitor timer (L) (H)	Text (command)
(14 bytes)	(20 bytes)	(8 bytes)	(2 bytes)	(2 bytes)	(4 bytes)	(Maximum 2040 bytes)

Differs depending on the function (application data portion)

(b) Order during response reception

Header			Application data (*1 *2)		
Ethernet	IP	UDP	Subheader (H) (L)	End code (H) (L)	Text (response)
(14 bytes)	(20 bytes)	(8 bytes)	(2 bytes)	(2 bytes)	

Differs depending on the function (application data portion)

*1 The data order for each function and the data order when the status is normal are shown in each function explanation item from Item 10.2 and later.

*2 The application data portion data order is as follows when the response end code is "5" "B" (error end).

Header	Application data			
	Subheader (H) (L)	End code (H) (L) "5" "B"	Error code (H) (L)	"0" "0"
	(2 bytes)	(2 bytes)	(2 bytes)	(2 bytes)

10.1.3 Exchange Data Item Contents

The following shows the command and response data item contents when conducting exchange by reading/writing data in the PLC CPU.

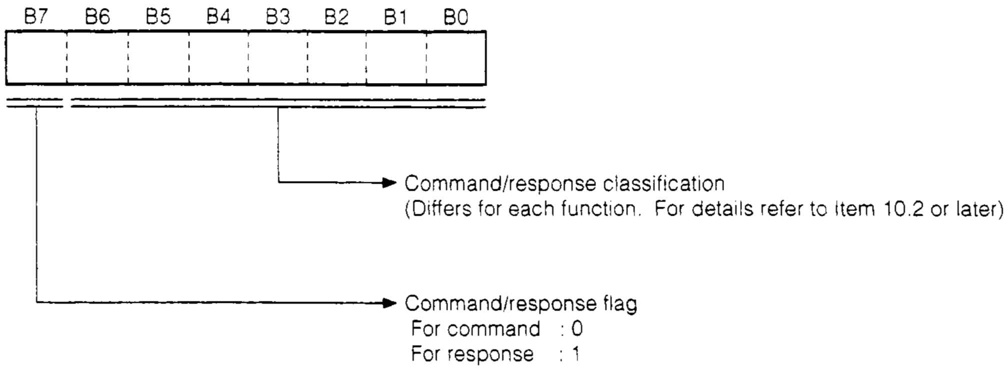
For the response returned to the remote node by the E71, the E71 automatically sets the data, so it is not necessary for the user to make the setting.

1 Header

The header is the header used by the TCP/IP or UDP/IP. For the E71, it is added or removed by the E71, so the user is not required to do the setting.

2 Subheader

The subheader format has the configuration shown below.



3 PLC No.

This shows for which PLC station the remote node conducts exchange by reading/writing data in the PLC CPU. Specify the target PLC station's PLC No. in accordance with Item 9.1.2 2 and Item 9.1.3 2.

(a) When exchanging using binary code the PLC No. is shown using a binary value.

(b) When exchanging using ASCII code the PLC No. is shown using ASCII code when expressed using a hexadecimal number.

(c) An example specification is shown at the end of this item (remarks).

4 ACPU monitor timer

The wait time from when the E71 (which has received request data from the other node) outputs a read/write request to the PLC CPU until the response is returned is indicated by the following value:

0000H (0)	: Unlimited wait
0001H to FFFFH (1 to 65535)	: Wait time (units 250ms)

(a) When exchange is done using binary code, the ACPU monitor time is shown using a binary value.

(b) When exchange is conducted using ASCII code, the ACPU monitor time is shown using ASCII code when expressed as a hexadecimal number.

(c) An example specification is shown at the end of item (remarks).

5 **Test (command)**

The E71 commands etc., that show the functions that can be used when a remote node reads/writes data in the PLC CPU in the target PLC station. The data contents and order for the text (command) portion differs depending on the functions used. The data order for each function is given in the function explanations from Item 10.2 and later.

6 **Text (response)**

This shows the read data/processing results etc., when a remote node reads/writes data in the PLC CPU in the target PLC station. The data contents and order of the text (response) portion varies depending on the functions used. The data order during normal end for each function is shown in the function explanations in Item 10.2 and later.

7 **End code**

The following values are used to show the processing results when a remote node reads/ writes data in the PLC CPU to a target PLC station.

00H : Normal end

Other than 00H : Error end (01H to B001H)

(a) When exchanging using binary code, the end code is shown as a binary value.

(b) When exchanging using ASCII code, the end code is shown as an ASCII code when expressed as a hexadecimal number.

(c) When an error end occurs, check the contents and take countermeasures in accordance with Chapter 13. When the end code is $5B_H$/"5B", the error code (10H to 21H) data immediately following and 00_H/"00" are included.

8 **Error code**

This shows the error contents when the end code is $5B_H$/"5B" when the processing result is an error when a remote node reads/writes data in the PLC CPU to the target PLC station. (Error code: 10H to 21H)

(a) When exchanging using binary code, the fault code is shown as a binary value.

(b) When exchanging using ASCII code, the fault code is shown as an ASCII code when expressed as a hexadecimal number.

(c) Check the contents and conduct countermeasures in accordance with Chapter 13.

The data code(ASCII, binary) used for transmission and reception of commands and responses between the E71 and a remote node are set using the data code setting switch (SW2) on the front of the E71.

Transmit the values handled by the items in the command and response by the E71 and the remote node to which exchange is being conducted using the following codes in accordance with the above settings. In addition, conduct reception using the following codes. In the explanations for items hereafter, the values handled by the items in the command and responses will be shown as binary values.

(1) For Binary Code Exchange
Unless otherwise explained, the values shown in the explanations are the binary values and are transmitted and received in the specification order (L to H).

(2) For ASCII Code Exchange
Unless especially explained, the values given in the explanations are converted into hexadecimal ASCII code and transmitted and received in the specification order (H to L).

Remarks

Following shows an example specification of the subheader to ACPU monitor timer when data is read/written in the PLC CPU under the following conditions.

Specified value

- Target station : PLC CPU station installed in the E71 (Local station) .... FFH
- Function used : Device memory batch read (Bit unit) .... 00H (E71 command)
- ACPU monitor timer value : 2500ms: 000AH

1 Format when exchanging using binary code

(a) Order during command transmission (Remote node → E71)

Header	Application data											
	Subheader	PLC No.	ACPU monitor timer		Text (command)							
					(Head device No.)				(Device name)		Number of device points	
			(L)	(H)	(L)	–	–	(H)	(L)	(H)		
	00H	FFH	0AH	00H	64H	00H	00H	00H	20H	4DH	08H	00H
		(Local station)	(2500ms)		(100)				(M)		(8 points)	

(b) Order during response reception (Remote node ← E71)

Header	Application data					
	Subheader	End code	Text (response) Specified device's on/off status			
	80H	00H	10H	10H	01H	10H
		(Normal end)				

M100(ON)
M101(OFF)
M102(ON)
M103(OFF)
M104(OFF)
M105(ON)
M106(ON)
M107(OFF)

2 Format when exchanging using ASCII code

(a) Order during command transmission (Remote node → E71)

Header	Application data							
	Subheader		PLC No.		ACPU monitor timer			
	(H)	(L)	(H)	(L)	(H)			(L)
	"0"	"0"	"F"	"F"	"0"	"0"	"0"	"A"
	30H	30H	46H	46H	30H	30H	30H	41H
			(Local station)		(2500ms)			

Application data															
Text (command)															
(Device name)				(Head device No.)								Number of device points			
(H)	–	–	(L)	(H)	–	–	–	–	–	–	(L)	(H)	(L)		
"4"	"D"	"2"	"0"	"0"	"0"	"0"	"0"	"0"	"0"	"6"	"4"	"3"	"8"	"0"	"0"
34H	44H	32H	30H	30H	30H	30H	30H	30H	30H	36H	34H	30H	38H	30H	30H
(M)				(100)								(8 points)			

(b) Order during response reception (Remote node ← E71)

Header	Application data											
	Subheader		End code		Text (response) Specified device's on/off status							
	(H)	(L)	(H)	(L)	(H)	–	–	–	–	–	–	(L)
	"8"	"0"	"0"	"0"	"1"	"0"	"1"	"0"	"0"	"1"	"1"	"0"
	30H	30H	30H	30H	31H	30H	31H	30H	30H	31H	31H	30H
			(Normal end)		M100(ON)	M101(OFF)	M102(ON)	M103(OFF)	M104(OFF)	M105(ON)	M106(ON)	M107(OFF)

10.1.4 Thinking Regarding Transmission Data

This section explains the thinking regarding the transmission data that handles the character portions when data is transmitted and received between a remote node and the PLC using commands.

1 When Exchanging data using binary code

(a) When reading to and writing from bit device memories

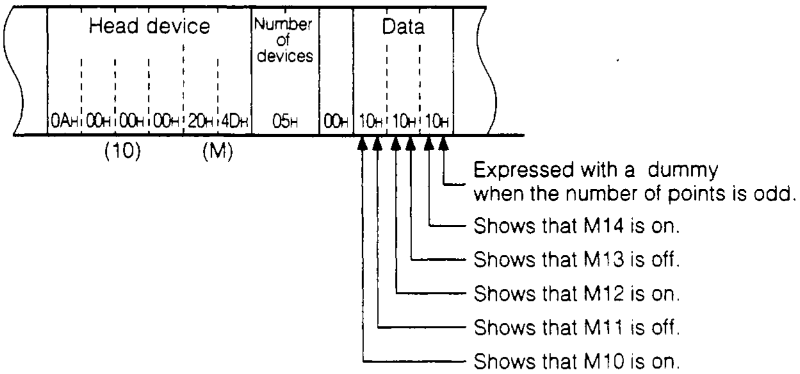
The bit device memory is sometimes handled in bit units (1 point units) and word units (16 points).

This section explains the thinking regarding these transmission data.

① Bit unit (1 point unit)

When bit device memory is handled in bit units, 1 point is specified as 4 bits and if the specified device number of points portions from the specified head device are turned on in the order from the first bit, then "1" is displayed and if off, then "0" is displayed.

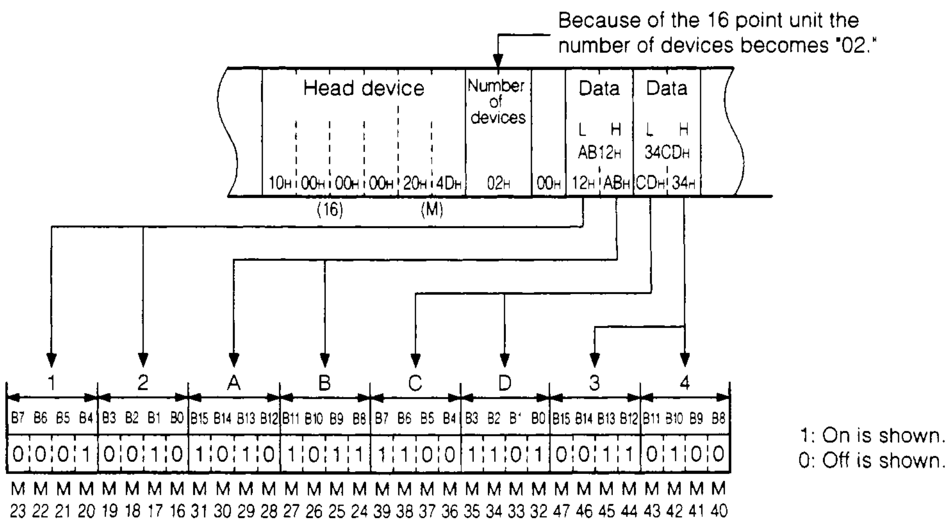
Example: When the 5 points from M10 are displayed in on/off.



② Word unit (16 point unit)

When the bit device memory is handled in 1 word units, 1 point is specified as 1 bit, and the specified device number of points from the specified head device is 16 point units, so the bits are expressed in the order from Low bytes (L: bits 0 to 7) to the High bytes (H: bits 8 to 15).

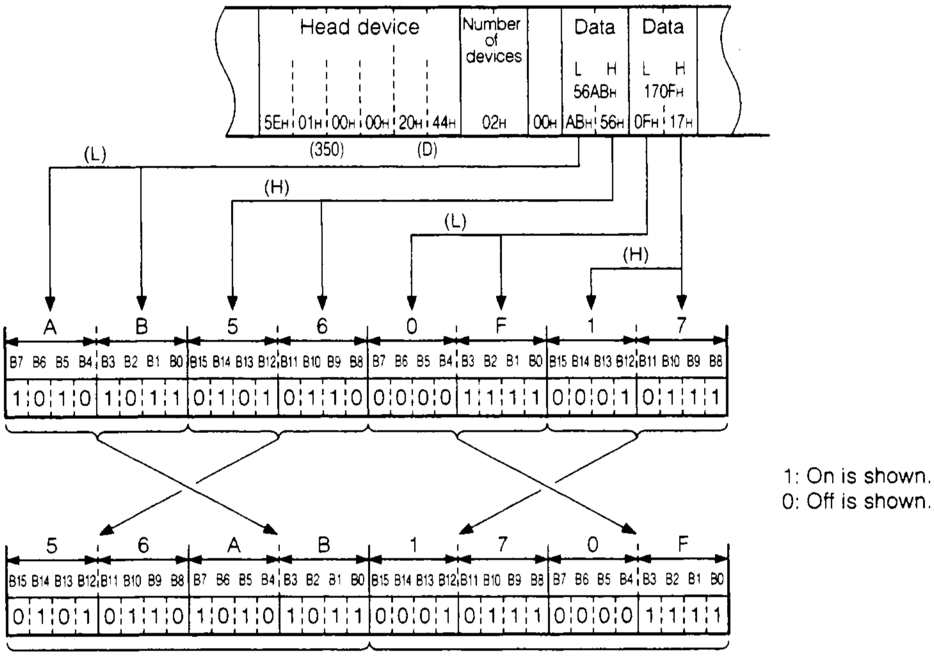
Example: When the 32 points from M16 are displayed in on/off.



(b) When reading to and writing from the word device memory

The word device memory is specified in one word as 16 bits, so the specified device number of bits from the specified head device is in 1 bit units, and the bits are displayed in the order from the Low bytes (L: bits 0 to 7) to the High bytes (H: bits 8 to 15).

Example: When the stored contents in data registers D350 and D351 are displayed.



The D350 contents show 56ABH (22187 in decimal).

The D351 contents show 170FH (5903 in decimal).

Point

When other than integers (real numbers, character strings) are stored in the word device memory that will read the data, the E71 reads the stored values as integer values.

Example: When real numbers (0.75) are stored in D0 to D1, the following integer values are read.
D0 = 0000H, D1 = 3F40H

Example 2: When character strings ("12AB") are stored in D2 to D3, the following integer values are read.
D2 = 3231H, D3 = 4241H

Remarks

The same thinking as used for the word device memory also applies to word data that handles buffer memory reading and writing.

2 When exchanging data using ASCII code

(a) When reading to and writing from bit device memory

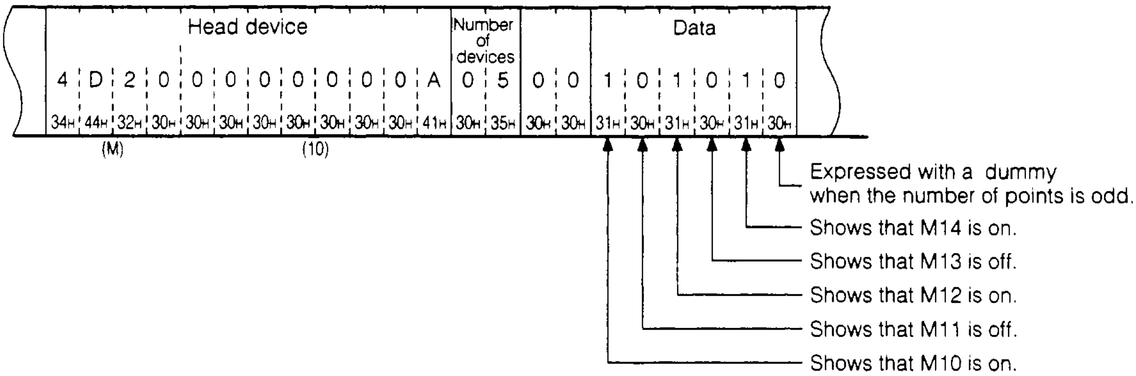
The bit device memory is sometimes handled in bit units (1 point units) and word units (16 points).

The following explains the thinking regarding the various transmission data.

① Bit unit (1 point unit)

When the bit device memory is handled in bit units, and if the specified device number of points portion from the specified head device are turned on in order from the left, then "1" (31H) is displayed and if off, then "0" (30H) is displayed.

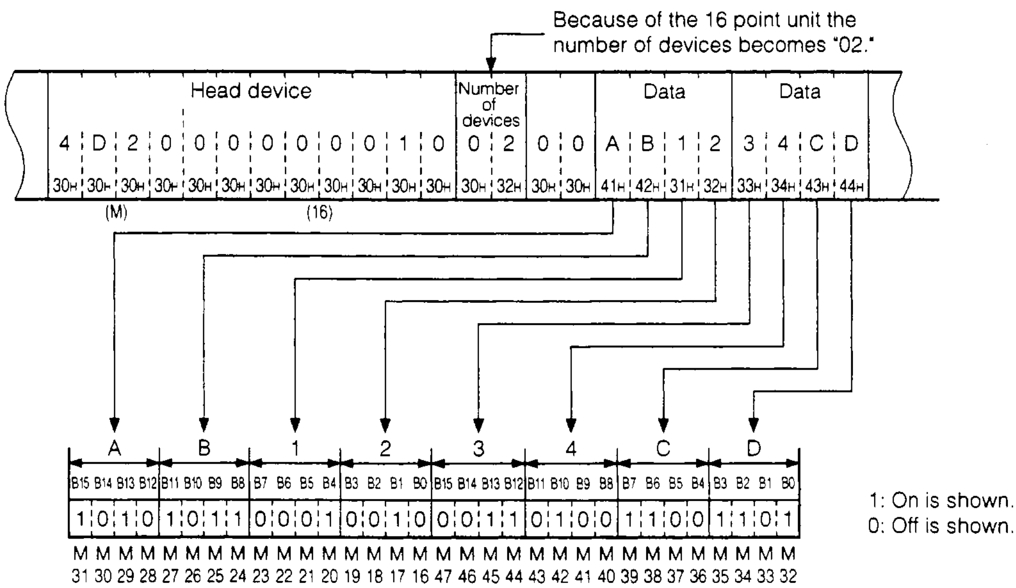
Example: When the 5 points from M10 are displayed in ON/OFF.



② Word units (16 point units)

When the bit device memory is handled in word units, one word is 4 bit units and the word is displayed in order from the first bit using hexadecimal numbers.

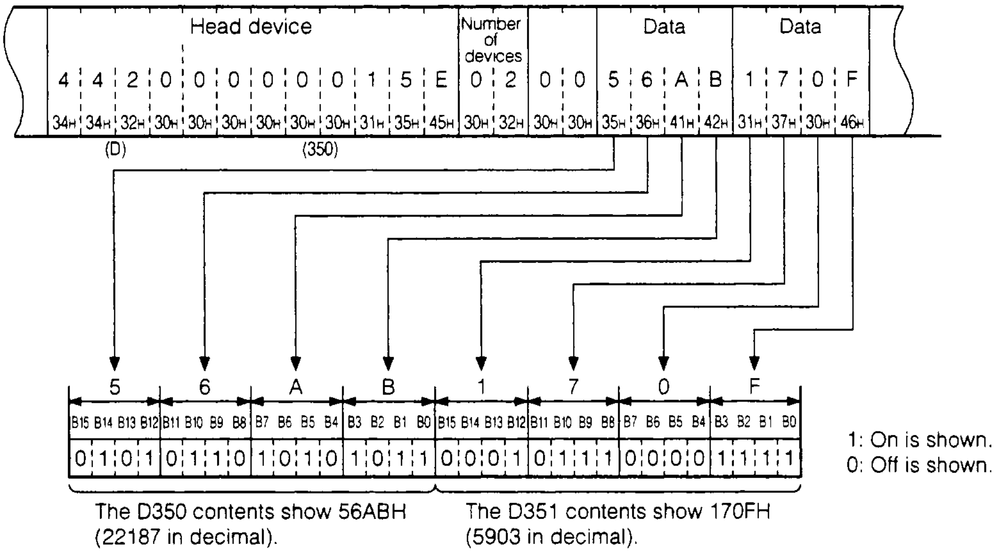
Example: When the 32 points from M16 are displayed in on/off.



(b) When reading to and writing from word device memory

The word device memory is one word of 4 bit units which are displayed from the first bit in order using hexadecimal numbers.

Example: When the stored contents in data registers D350 and D351 are displayed.



Point

(1) Use the capital character codes when alpha characters are specified for the text.

(2) When other than integers (real numbers, character strings) are stored in the word device memory that will read the data, the E71 reads the stored values as integer values.

Example: When real numbers (0.75) are stored in D0 to D1, the following integer values are read.
D0 = 0000H, D1 = 3F40H

Example 2: When character strings ("12AB") are stored in D2 to D3, the following integer values are read.
D2 = 3231H, D3 = 4241H

Remarks

The same thinking as used for the word device memory also applies to word data that handles buffer memory reading and writing.

10.2 Device Memory Read/Write

This section explains the control method for reading from and writing to the device memory.

10.2.1 Command and Device Range

1 **The functions occurring in device memory read and write are shown in Table 10.1.**

Table 10.1 Function list

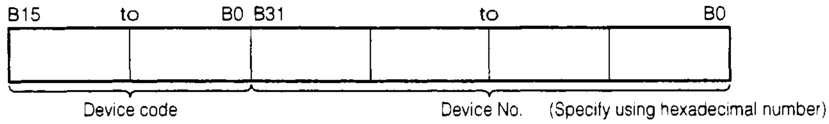
Item		Command/ response classification	Processing description	Number of processing points conducted in one communication	PLC CPU status		
					Stopped	Running	
						Write possible setting	Write impossible setting
Batch read	Bit unit	00_H	Bit devices (X, Y, M, etc.) are read in 1 point units	256 points	○	○	○
	Word unit	01_H	Bit devices (X, Y, M, etc.) are read in 16 point units	128 words (2048 points)			
			Word devices (D, R, T, C, etc.) are read in 1 points units	256 points			
Batch write	Bit unit	02_H	Bit devices (X, Y, M, etc.) are written to in 1 point units	256 points	○	○	×
	Word unit	03_H	Bit devices (X, Y, M, etc.) are written to in 16 point units	40 words (640 points)			
			Word devices (D, R, T, C, etc.) are read in 1 points units	256 points			
Test (random write)	Bit unit	04_H	The device and device No. for bit devices (X, Y, M, etc.) are randomly specified as set/reset in 1 points units.	80 points	○	○	×
	Word unit	05_H	The device and device No. for bit devices (X, Y, M, etc.) are randomly specified as set/reset in 16 points units.	40 words (640 points)			
			The device and device No. of word devices (D, R, T, C, etc.) are randomly specified as write in 1 point units.	40 points			
Monitor data registration	Bit unit	06_H	The bit devices to be monitored (X, Y, M, etc.) are set in 1 point units.	40 points *	○	○	○
	Word unit	07_H	The bit devices to be monitored (X, Y, M, etc.) are set in 16 point units.	20 words * (320 points)			
			The word devices to be monitored (D, R, T, C, etc.) are set in 1 point units.	20 points			
Monitor	Bit unit	08_H	Monitors the device that conducted the monitor data registration.	(Number of registrations)	○	○	○
	Word unit	09_H					

In the PLC CPU status column in the above table the "O" represents execution possible and the "X" represents execution not possible. The *number for when other than AnA, AnU, and QnA, 2 points are processed for each point for the device X (input). For example, when X is included in the specified device in monitor data registration bit units, make it so that

((X specified number of points) x 2 + other device specified number of points) <= number of points processed during one exchange

2 Device specification method and range

(a) The device setting method for device memory read/write is performed using the device code and device No. shown in the diagram below.



Example: For D100

D100=4420 00000064H

→ Device No. (hexadecimal)

→ Device code

(b) The device codes and device Nos. are shown in Table 10.2.

Table 10.2 Device List (CPU module without restrictions)

○ : Access enabled × : Access disabled — : No device

Device (*1)		Device code	Device range (*1)	Device No.	A1S, A1SH, A1SJ, A1SJH, A1, A1N	A2S, A2SH A2, A2N, A2C, A2CJ, A0J2H	A2-S1, A2N-S1 A2SH-S1	A3, A3N	A2A	A2A-S1	A3A
Data register		D0 (44H, 20H)	D0 to D1023	0000H to 03FFH	○	○	○	○	○	○	○
			D1024 to D6143	0400H to 17FFH	—	—	—	—	○	○	○
			D9000 to D9255	2328H to 2427H	○	○	○	○	○	○	○
Link register		W0 (57H, 20H)	W0 to W3FF	0000H to 03FFH	○	○	○	○	○	○	○
			W400 to WFFF	0400H to 0FFFH	—	—	—	—	○	○	○
File register		R0 (52H, 20H)	R0 to R4095	0000H to 0FFFH	—	○	○	○	○	○	○
			R4096 to R8191	1000H to 1FFFH	—	—	—	○	○	○	○
Timer	Current value	TN (54H, 4EH)	T0 to T255	0000H to 00FFH	○	○	○	○	○	○	○
			T256 to T2047	0100H to 07FFH	—	—	—	—	○	○	○
	Contact	TS (54H, 53H)	T0 to T255	0000H to 00FFH	○	○	○	○	○	○	○
			T256 to T2047	0100H to 07FFH	—	—	—	—	○	○	○
	Coil	TC (54H, 43H)	T0 to T255	0000H to 00FFH	○	○	○	○	○	○	○
			T256 to T2047	0100H to 07FFH	—	—	—	—	○	○	○
Counter	Current value	CN (43H, 4EH)	C0 to C255	0000H to 00FFH	○	○	○	○	○	○	○
			C256 to C1023	0100H to 03FFH	—	—	—	—	○	○	○
	Contact	CS (43H, 53H)	C0 to C255	0000H to 00FFH	○	○	○	○	○	○	○
			C256 to C1023	0100H to 03FFH	—	—	—	—	○	○	○
	Coil	CC (43H, 43H)	C0 to C255	0000H to 00FFH	○	○	○	○	○	○	○
			C256 to C1023	0100H to 03FFH	—	—	—	—	○	○	○
Input		X0 (58H, 20H)	X0 to X0FF	0000H to 00FFH	○	○	○	○	○	○	○
			X100 to X1FF	0100H to 01FFH	—	○	○	○	○	○	○
			X200 to X3FF	0200H to 03FFH	—	—	○	○	—	○	○
			X400 to X7FF	0400H to 07FFH	—	—	—	○	—	—	○
Output		Y0 (59H, 20H)	Y0 to Y0FF	0000H to 00FFH	○	○	○	○	○	○	○
			Y100 to Y1FF	0100H to 01FFH	—	○	○	○	○	○	○
			Y200 to Y3FF	0200H to 03FFH	—	—	○	○	—	○	○
			Y400 to Y7FF	0400H to 07FFH	—	—	—	○	—	—	○
Internal relay * Latch relay and step relay are included		M0 (4DH, 20H)	M0 to M2047	0000H to 07FFH	○	○	○	○	○	○	○
			M2048 to M8191	0800H to 1FFFH	—	—	—	—	○	○	○
			M9000 to M9255	2328H to 2427H	○	○	○	○	○	○	○
Link relay		B0 (42H, 20H)	B0 to B3FF	0000H to 03FFH	○	○	○	○	○	○	○
			B400 to BFFF	0400H to 0FFFH	—	—	—	—	○	○	○
Annunciator		F0 (46H, 20H)	F0 to F255	0000H to 00FFH	○	○	○	○	○	○	○
			F256 to F2047	0100H to 07FFH	—	—	—	—	○	○	○

Table 10.2 Device List (CPU module with restrictions)

○ : Access enabled × : Access disabled — : No device

Device (*1)		Device code	Device range (*1)	Device No.	A2AS, A2U	A2AS-S1, A2U-S1	A3U, A4U	Q2A, Q2AS, Q2ASH	Q2A-S1, Q2AS-S1, Q2ASHS1	Q3A, Q4A, Q4AR
Data register		D0 (44_H, 20_H)	D0 to D6143	0000H to 17FFH	○			○		
			D6144 to D8191	1800H to 1FFFH	×			×		
			D8192 or more	2000H or more	—			×		
			D9000 to D9255 (SD1000 to SD1255)	2328H to 2427H	○			○		
			(SD1256 to SD2047)		—			×		
Link register		W0 (57_H, 20_H)	W0 to WFFF	0000H to 0FFFH	○			○		
			W1000 to W1FFF	1000H to 1FFFH	×			×		
			W2000 or more	2000H or more	—			×		
File register		R0 (52_H, 20_H)	R0 to R8191	0000H to 1FFFH	○			×		
			R8192 or more	2000H or more	—			×		
Timer	Current value	TN (54_H, $4E_H$)	T0 to T2047	0000H to 07FFH	○			○		
			T2048 or more	0800H or more	—			×		
	Contact	TS (54_H, 53_H)	T0 to T2047	0000H to 07FFH	○			○		
			T2048 or more	0800H or more	—			×		
	Coil	TC (54_H, 43_H)	T0 to T2047	0000H to 07FFH	○			○		
			T2048 or more	0800H or more	—			×		
Counter	Current value	CN (43_H, $4E_H$)	C0 to C1023	0000H to 03FFH	○			○		
			C1024 or more	0400H or more	—			×		
	Contact	CS (43_H, 53_H)	C0 to C1023	0000H to 03FFH	○			○		
			C1024 or more	0400H or more	—			×		
	Coil	CC (43_H, 43_H)	C0 to C1023	0000H to 03FFH	○			○		
			C1024 or more	0400H or more	—			×		
Input		X0 (58_H, 20_H)	X0 to X1FF	0000H to 01FFH	○			○		
			X200 to X3FF	0200H to 03FFH	×	○		×	○	
			X400 to X7FF	0400H to 07FFH	×		○	×		○
			X800 to X1FFF	0800H to 1FFFH	×			×		
			X2000 or more	2000H or more	—			×		
Output		Y0 (59_H, 20_H)	Y0 to Y1FF	0000H to 01FFH	○			○		
			Y200 to Y3FF	0200H to 03FFH	×	○		×	○	
			Y400 to Y7FF	0400H to 07FFH	×		○	×		○
			Y800 to Y1FFF	0800H to 1FFFH	×			×		
			Y2000 or more	2000H or more	—			×		
Internal relay		M0 ($4D_H$, 20_H)	M0 to M8191	0000H to 1FFFH	○ * Including Latch relay (L)/Step relay (S).			○		
			M8192 or more	2000H or more	—			×		
			M9000 to M9255 (SM1000 to SM1255)	2328H to 2427H	○			○		
			(SM1256 to SM2047)		—			×		
Latch relay					(Depends on the above)			× * Access to Internal relay (M) even if Latch relay (L) is specified.		
Step relay					(Depends on the above)			× * Access to Internal relay (M) even if Step relay (S) is specified.		
Link relay		B0 (42_H, 20_H)	B0 to BFFF	0000H to 0FFFH	○			○		
			B1000 to B1FFF	1000H to 1FFFH	×			×		
			B2000 or more	2000H or more	—			×		
Annunciator		F0 (46_H, 20_H)	F0 to F2047	0000H to 07FFH	○			○		
			F2048 or more	0800H or more	—			×		

*1 Refer to Item 9.2*4 for precaution items when reading/writing to and from the QnACPU.

Point

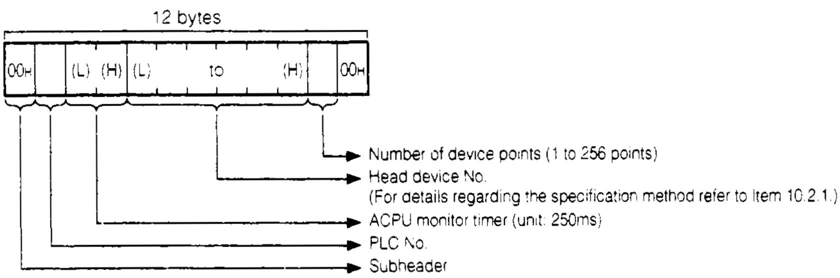
(1) The bit device and word device classifications are as follows.
Bit device......... X, Y, M, L, B, F, T (contact), T (coil), C (contact), C (coil)
Word device .... T (current value), C (current value), D, W, R
(2) Be sure to use device Nos. that are in multiples of 16 for the bit device's device No. for word unit specification.
(3) The special relay's (M9000 to M9255) and special registers (D9000 to D9255), are for special read, special write, and system use. Conducting a write in areas outside of the write possible range will cause a PLC CPU error. For details regarding the special relay and the special register, refer to the ACPU programming manual.
(4) When reading/writing file registers to and from the PLC CPU that uses extension file registers, use the commands explained in "Item 10.3 Extension File Register Reading/Writing." When using the extension file registers, there are times when correct reading and writing cannot be done when processing file registers using device batch read and write.
(5) Reading/writing to the AnUCPU and QnACPU can be done in the AnACPU device range.

10.2.2 Bit Unit Batch Read

This section explains the command/response format when conducting bit device memory batch read.

1 When exchanging using binary code

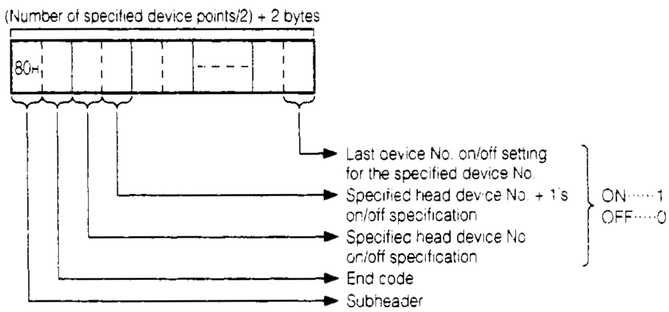
Command format



Remarks

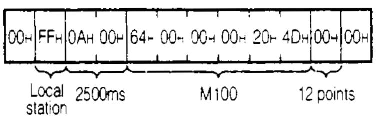
Set to "00H" when specifying the number of device points as 256 points.

Response format

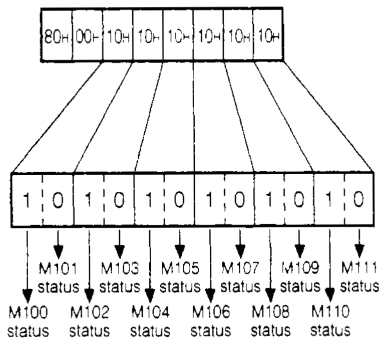


Example: When reading the M100 to M111 on/off status of the PLC CPU installed in E71.

Command (remote node → E71)

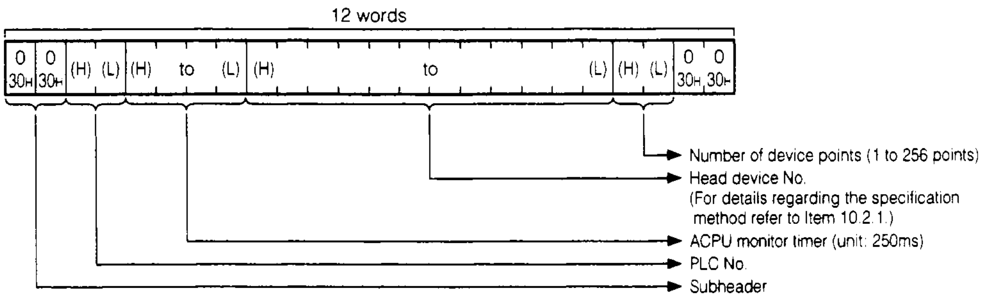


Response (E71 → remote node)



2 When exchanging using ASCII code

Command format

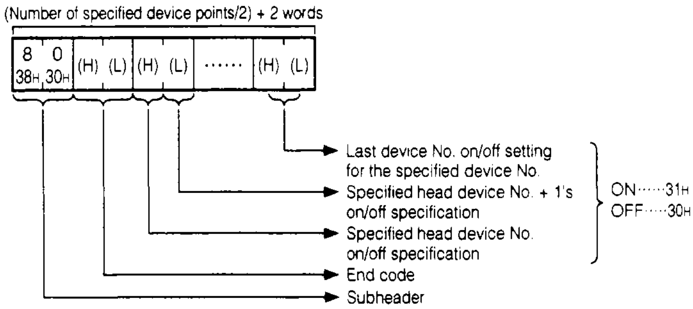


Remarks

(1) Set to "3030H" when specifying the number of device points as 256 points.

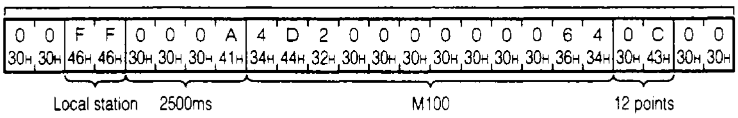
(2) 1 byte of dummy data (30H) is added to the response data when the number of specified device points is odd. For example, 4 points worth of data is returned when 3 points are read. The last 1 byte is the dummy data.

Response format

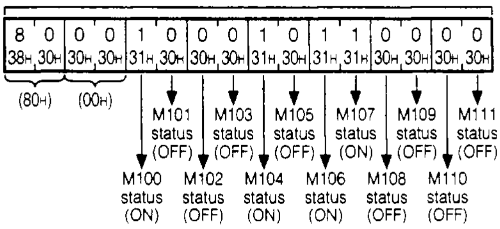


Example: When reading the M100 to M111 on/off status of the PLC CPU installed in E71.

Command (remote node → E71)



Response (E71 → remote node)

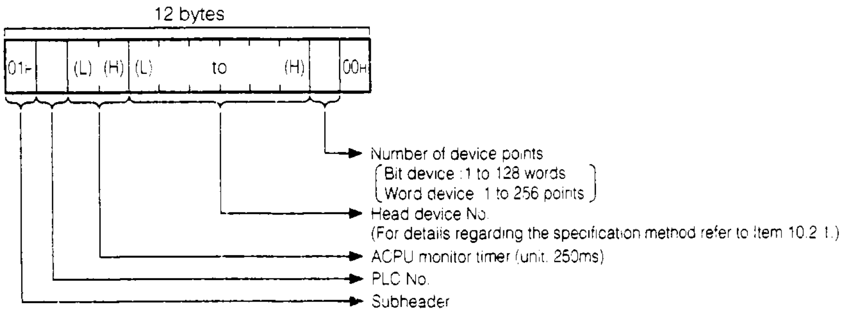


10.2.3 Word Unit Batch Read

This section explains the command/response format when conducting word device memory batch read and bit device memory (16 unit) batch read.

1 **When exchanging using binary code**

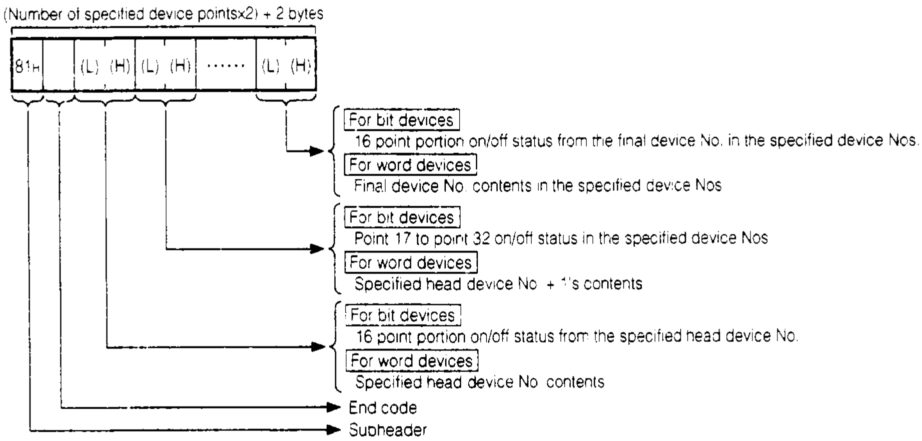
Command format



Remarks

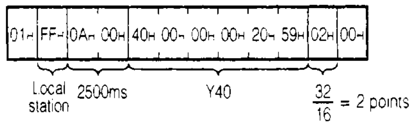
Set to "00H" when specifying the number of device points as 256 points.

Response format

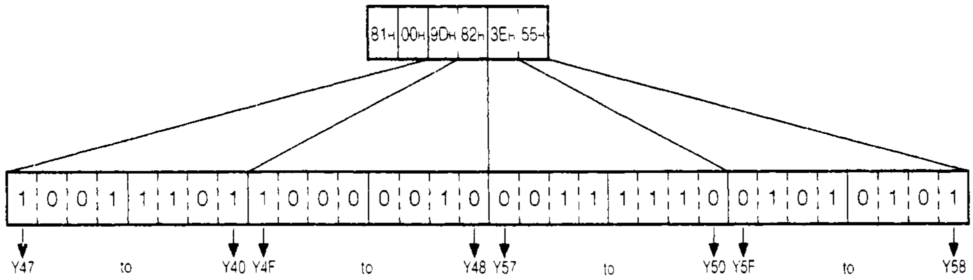


Example 1: When reading the Y40 to 5F (32 points) on/off status of the PLC CPU installed in E71.

Command (remote node → E71)

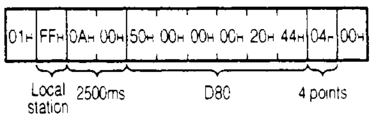


Response (E71 → remote node)

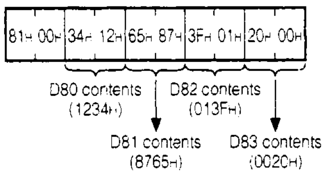


Example 2: When reading the D80 to 83 contents of the PLC CPU installed in E71.

Command (remote node → E71)

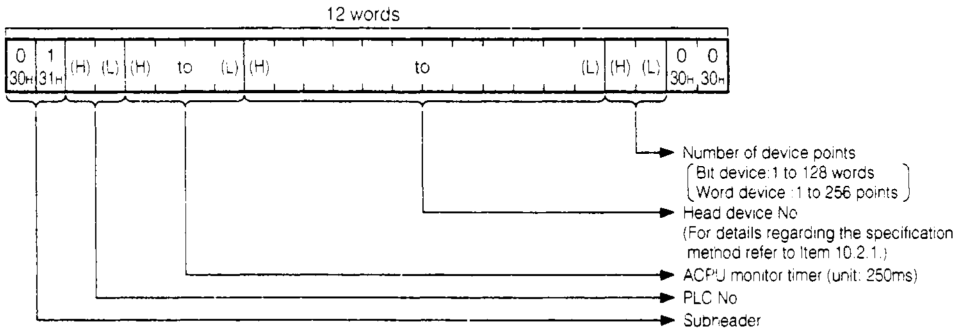


Response (E71 → remote node)



2 When exchanging using ASCII code

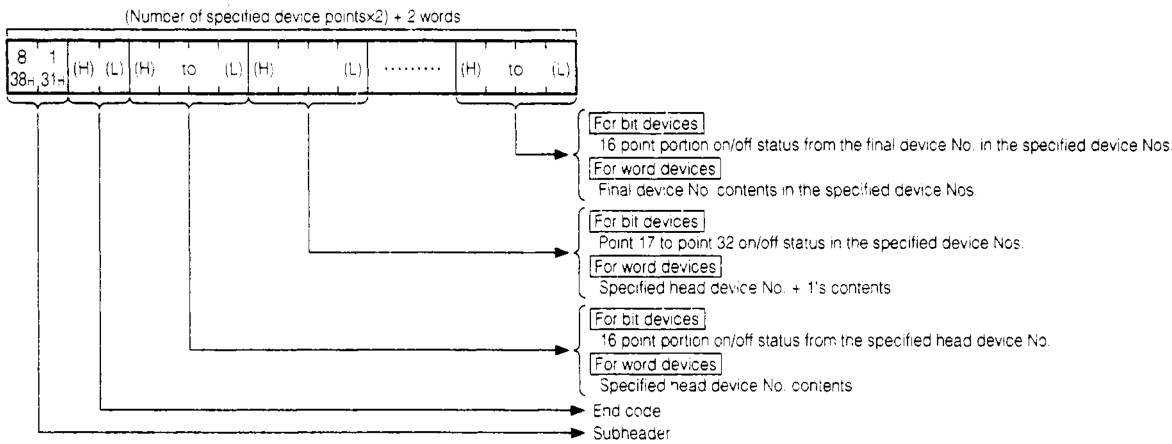
Command format



Remarks

Set to "3030H" when specifying the number of device points as 256 points.

Response format



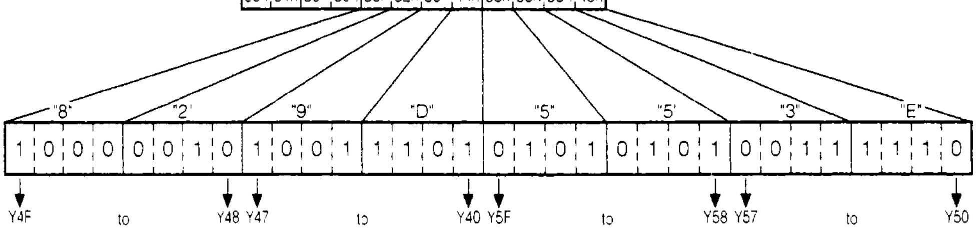
Example 1: When reading the Y40 to 5F (32 points) on/off status of the PLC CPU installed in E71.

Command (remote node → E71)

0	1	F	F	0	0	0	A	5	9	2	0	0	0	0	0	0	0	4	0	0	2	0	0
30H	31H	46H	46H	30H	30H	30H	41H	35H	39H	32H	30H	30H	30H	30H	30H	30H	30H	34H	30H	30H	32H	30H	30H
		Local station		2500ms				Y40												$\frac{32}{16}$ = 2 points			

Response (E71 → remote node)

8	1	0	0	8	2	9	D	5	5	3	E
38H	31H	30H	30H	38H	32H	39H	44H	35H	35H	33H	45H



Example 2: When reading the D80 to D82 contents of the PLC CPU installed in E71.

Command (remote node → E71)

0	1	F	F	0	0	0	A	4	4	2	0	0	0	0	0	0	0	5	0	0	3	0	0
30H	31H	46H	46H	30H	30H	30H	41H	34H	34H	32H	30H	30H	30H	30H	30H	30H	30H	35H	30H	30H	33H	30H	30H
		Local station		2500ms				D80												3 points			

Response (E71 → remote node)

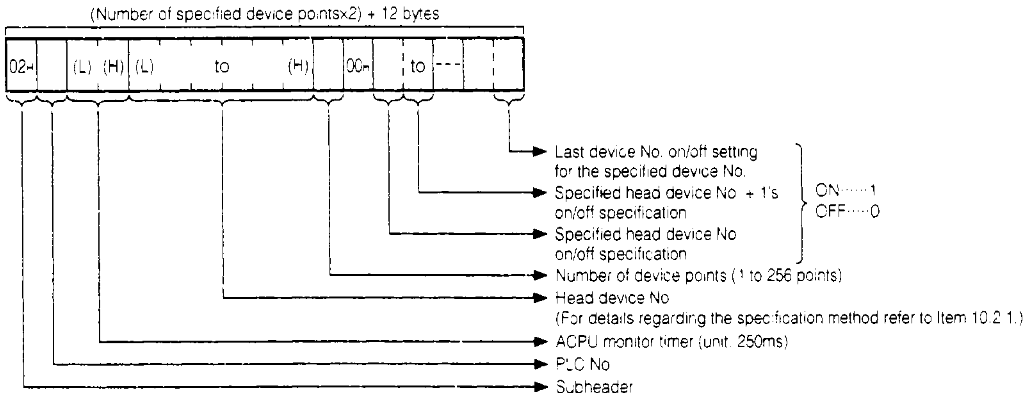
8	1	0	0	1	2	3	4	8	7	6	5	0	1	3	F
38H	31H	30H	30H	31H	32H	33H	34H	38H	37H	36H	35H	30H	31H	33H	46H
				D80 contents (1234H)				D81 contents (8765H)				D82 contents (013FH)			

10.2.4 Bit Unit Batch Write

This section explains the command/response format when conducting bit device memory batch write.

1 When exchanging using binary code

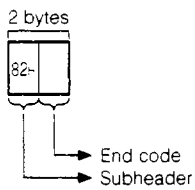
Command format



Remarks

Set to "00H" when specifying the number of device points as 256 points.

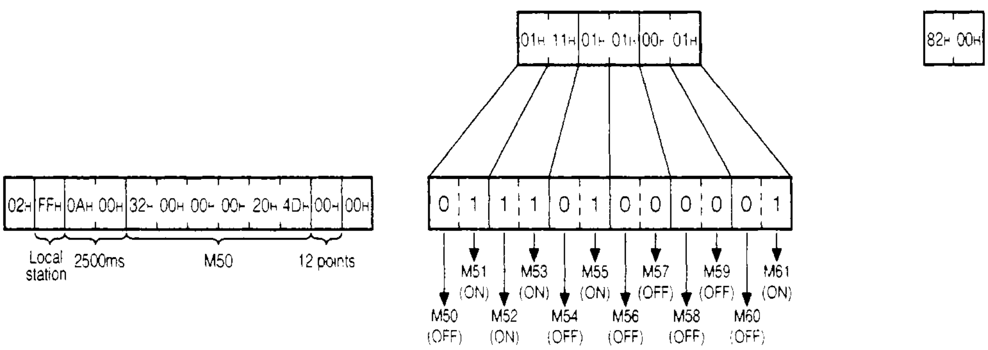
Response format



Example: When reading the M50 to M61 on/off status of the PLC CPU installed in E71.

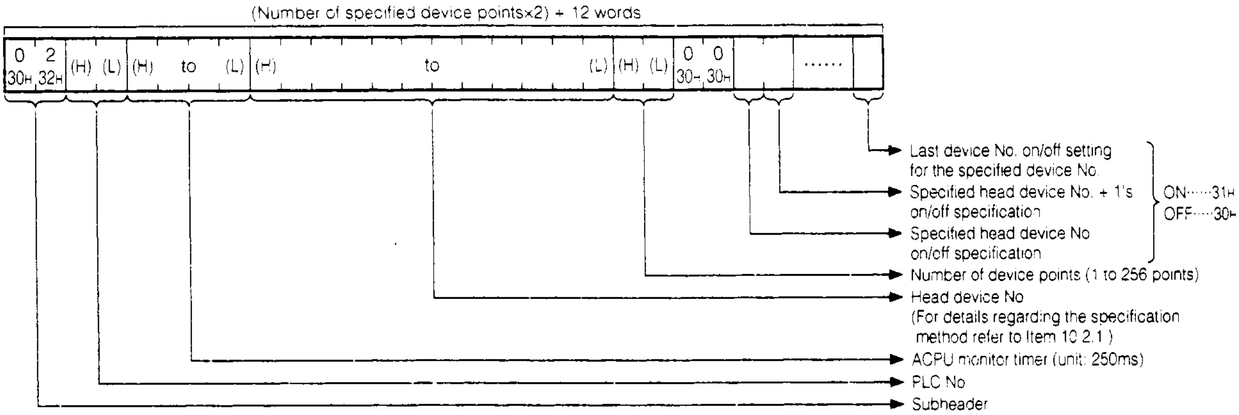
Command (remote node → E71)

Response (E71 → remote node)



2 When exchanging using ASCII code

Command format

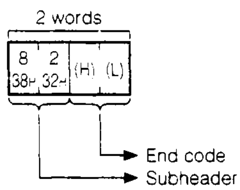


Remarks

(1) Set to "3030H" when specifying the number of device points as 256 points.

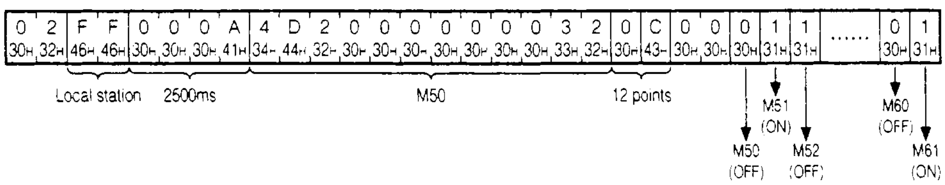
(2) 1 byte of dummy data (30H) is added to the end of the write data when the number of specified device points is odd. For example, when 3 points are read the dummy data (30H) is added to the end.

Response format

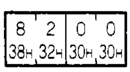


Example: When writing the M50 to M61 on/off data to the PLC CPU installed in E71.

Command (remote node → E71)



Response (E71 → remote node)

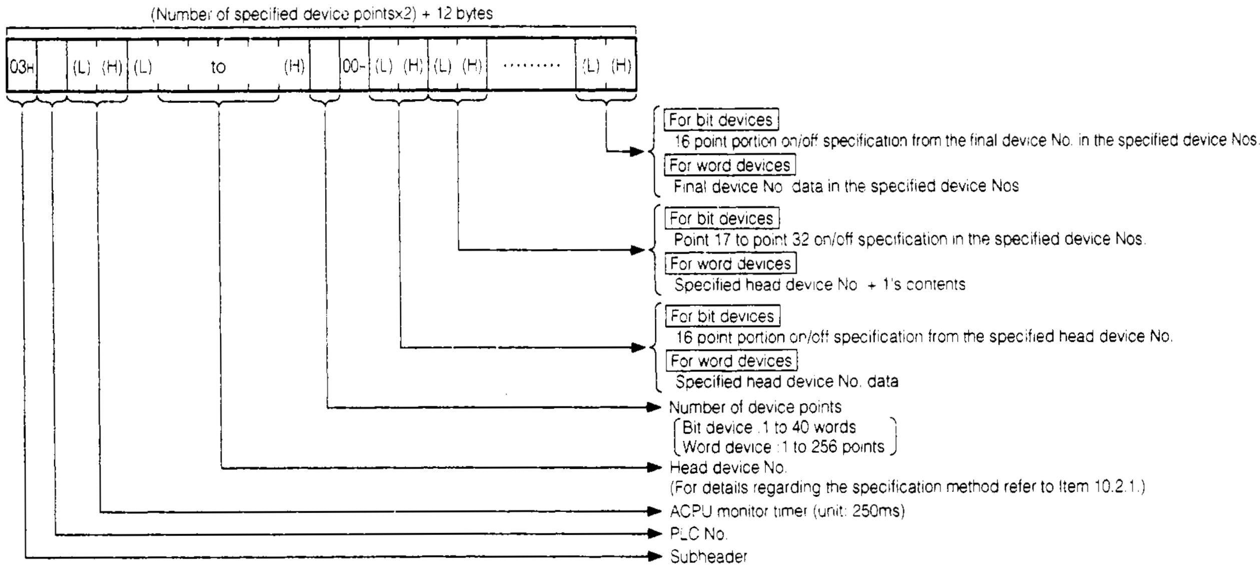


10.2.5 Word Unit Batch Write

This section explains the command/response format when conducting word device memory batch write and bit device memory (16 unit) batch write.

1 When exchanging using binary code

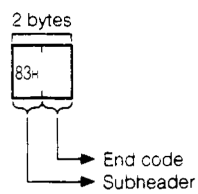
Command format



Remarks

Set to "00H" when specifying the number of device points as 256 points.

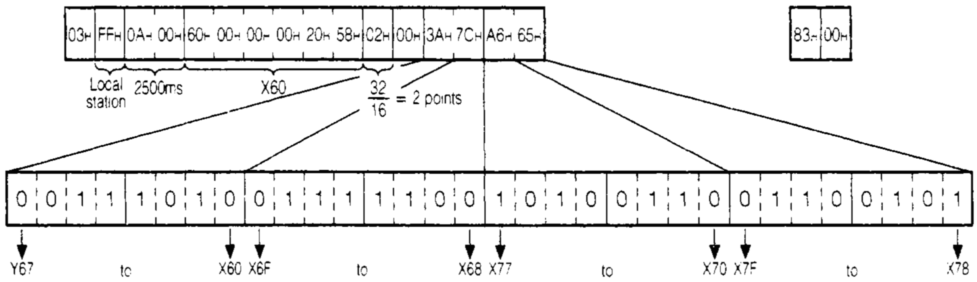
Response format



Example 1: When writing the X60 to 7F (32 points) on/off data to the PLC CPU installed in E71.

Command (remote node → E71)

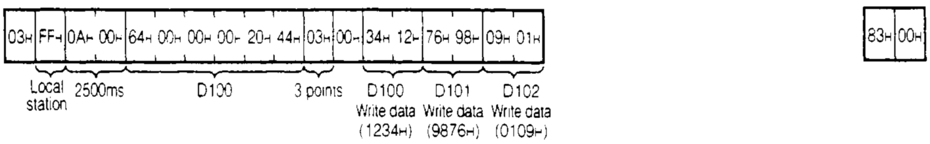
Response (E71 → remote node)



Example 2: When writing the D100 to 102 data to the PLC CPU installed in E71.

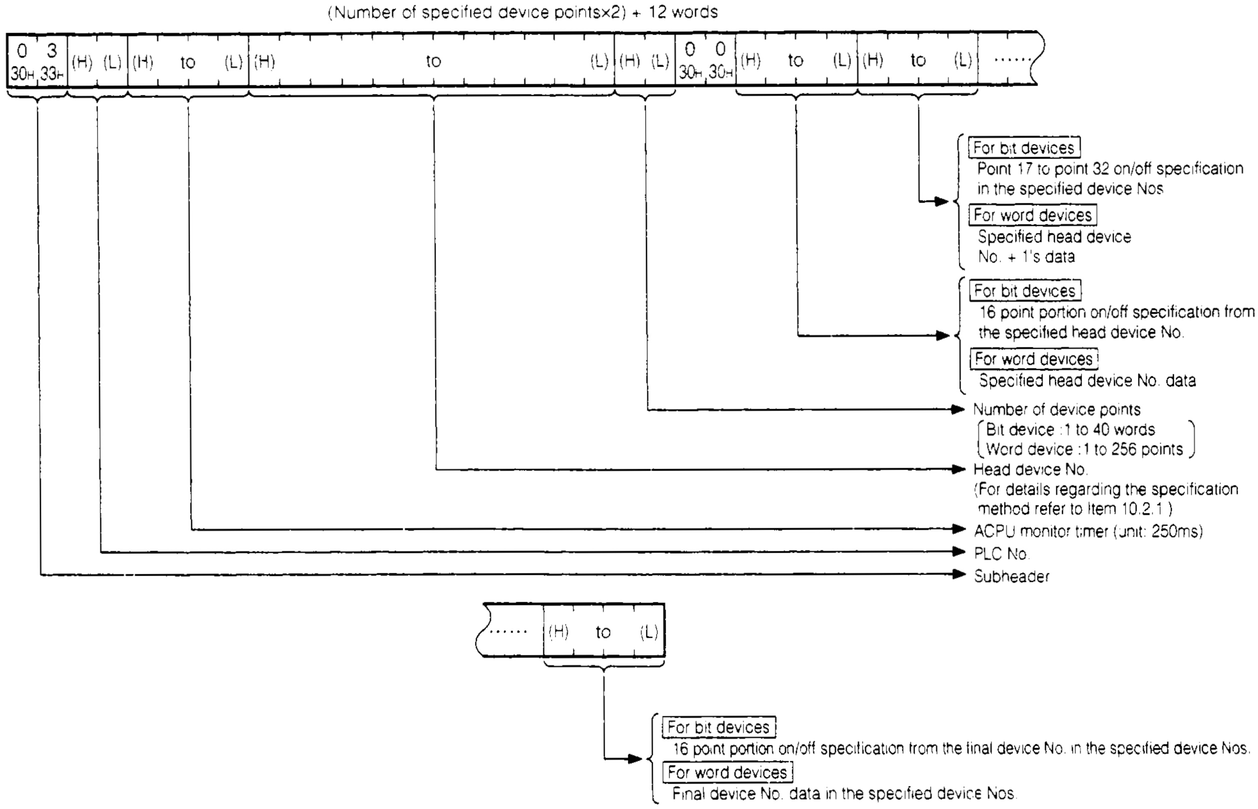
Command (remote node → E71)

Response (E71 → remote node)



2 When exchanging using ASCII code

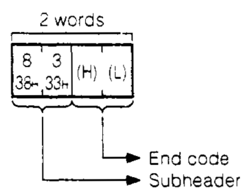
Command format



Remarks

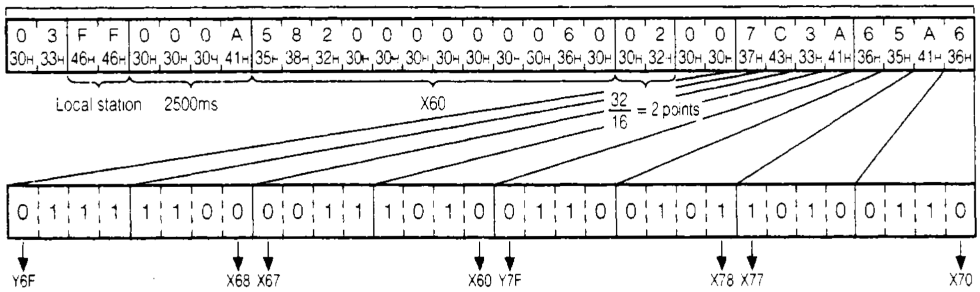
Set to "3030H" when specifying the number of device points as 256 points.

Response format



Example 1: When writing the X60 to 7F (32 points) on/off data to the PLC CPU installed in E71.

Command (remote node → E71)



Response (E71 → remote node)

8	3	0	0
38H	33H	30H	30H

Example 2: When writing the D100 to 102 data to the PLC CPU installed in E71.

Command (remote node → E71)

0	3	F	F	0	0	0	A	4	4	2	0	0	0	0	0	0	0	6	4	0	3	0	0	1	2	3	4	9	8	7	6	0	1	0	9
30H	33H	46H	46H	30H	30H	30H	41H	34H	34H	32H	30H	30H	30H	30H	30H	30H	30H	36H	34H	30H	33H	30H	30H	31H	32H	33H	34H	39H	38H	37H	36H	30H	31H	30H	39H
		Local station		2500ms				D100												3 points (03H)				D100 Write data (1234H)				D101 Write data (9876H)				D102 Write data (0109H)			

Response (E71 → remote node)

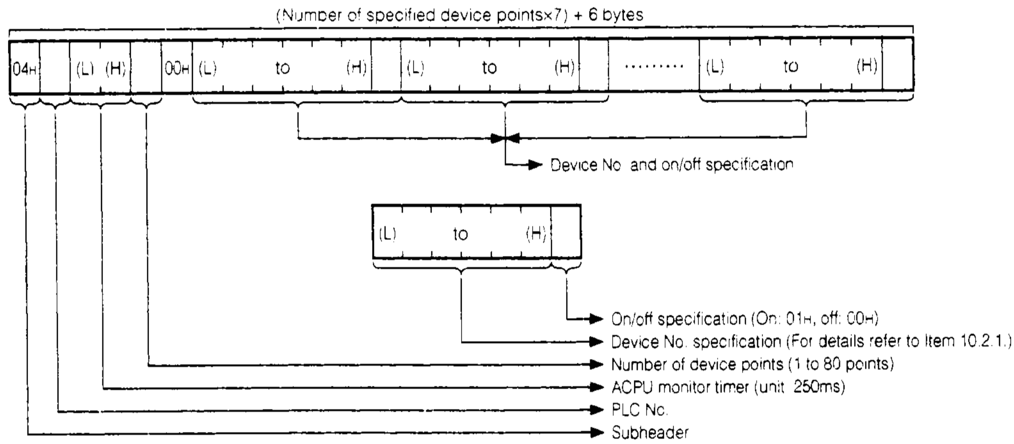
8	3	0	0
38H	33H	30H	30H

10.2.6 Bit Unit Test (Random Write)

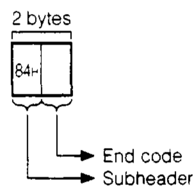
This section explains the command/response format when conducting a random write to a bit device memory.

1 When exchanging using binary code

Command format



Response format



Example: When X94 is turned on, the M60 is turned off, and the B26 is turned on for the PLC CPU installed in E71.

Command (remote node → E71)

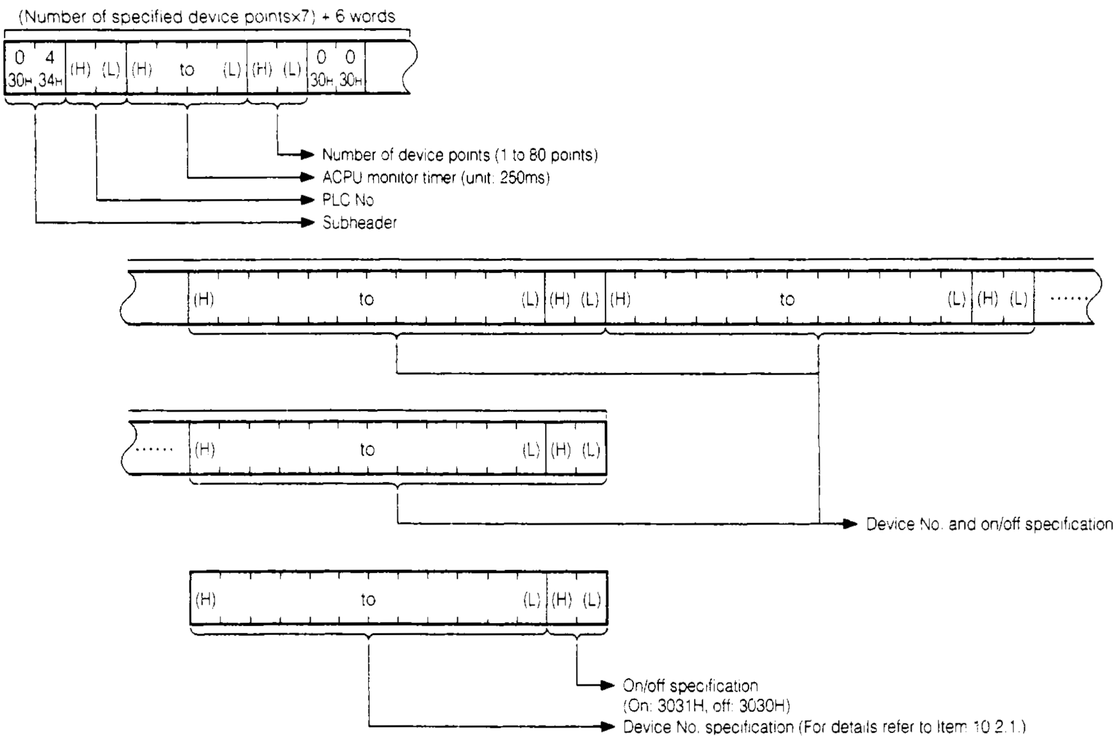
04H	FFH	0AH 00H	03H	00H	94H 00H 00H 00H 20H 58H	01H	3CH 00H 00H 00H 20H 4DH	00H	26H 00H 00H 00H 20H 42H	01H
	Local station	2500ms	3 points		X94	ON	M50	OFF	B26	ON

Response (E71 → remote node)

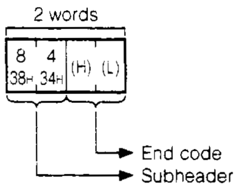
84H	00H

2 When exchanging using ASCII code

Command format

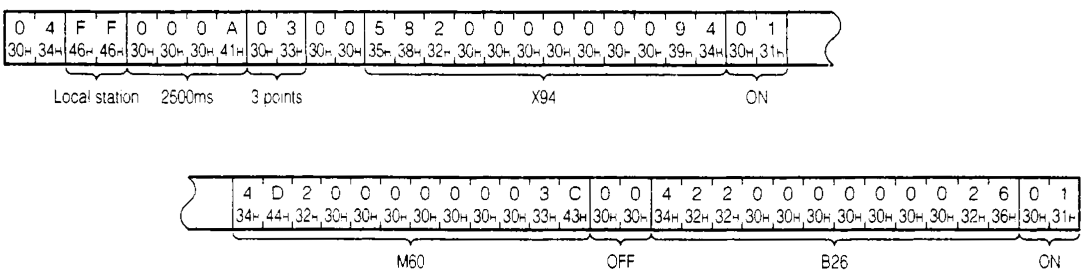


Response format

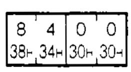


Example: When X94 is turned on, the M60 is turned off, and the B26 is turned on for the PLC CPU installed in E71.

Command (remote node → E71)



Response (E71 → remote node)

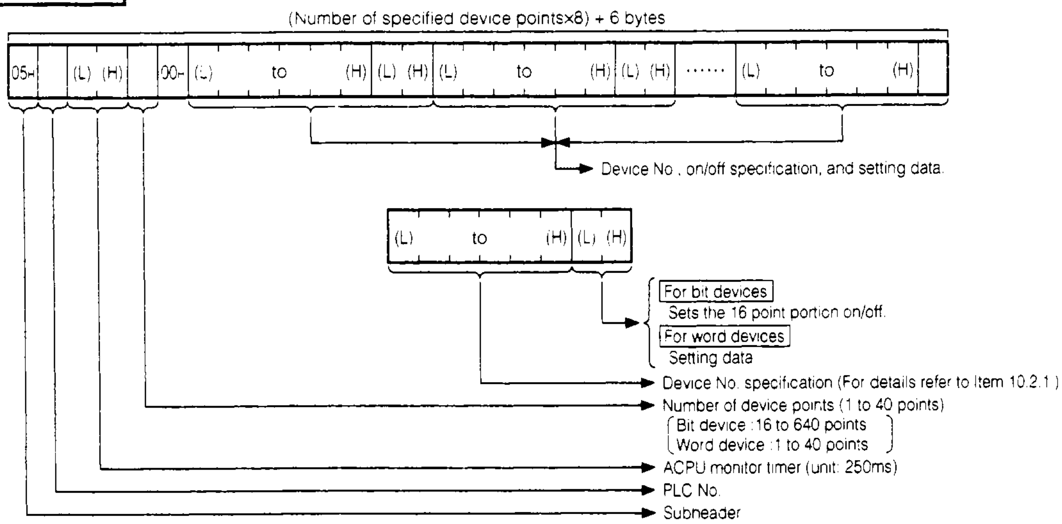


10.2.7 Word Unit Test (Random Write)

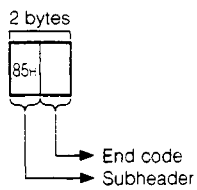
This section explains the command/response format when conducting a random write to a word device memory and bit device memory (16 point unit).

1 When exchanging using binary code

Command format

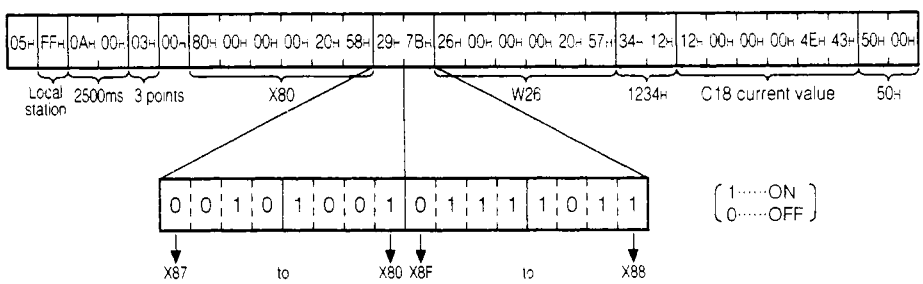


Response format

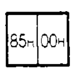


Example: When X80 to 8F is turned on/off and W26 is overwritten to "1234H" and C18's current value is overwritten to "50H" for the PLC CPU installed in E71.

Command (remote node → E71)

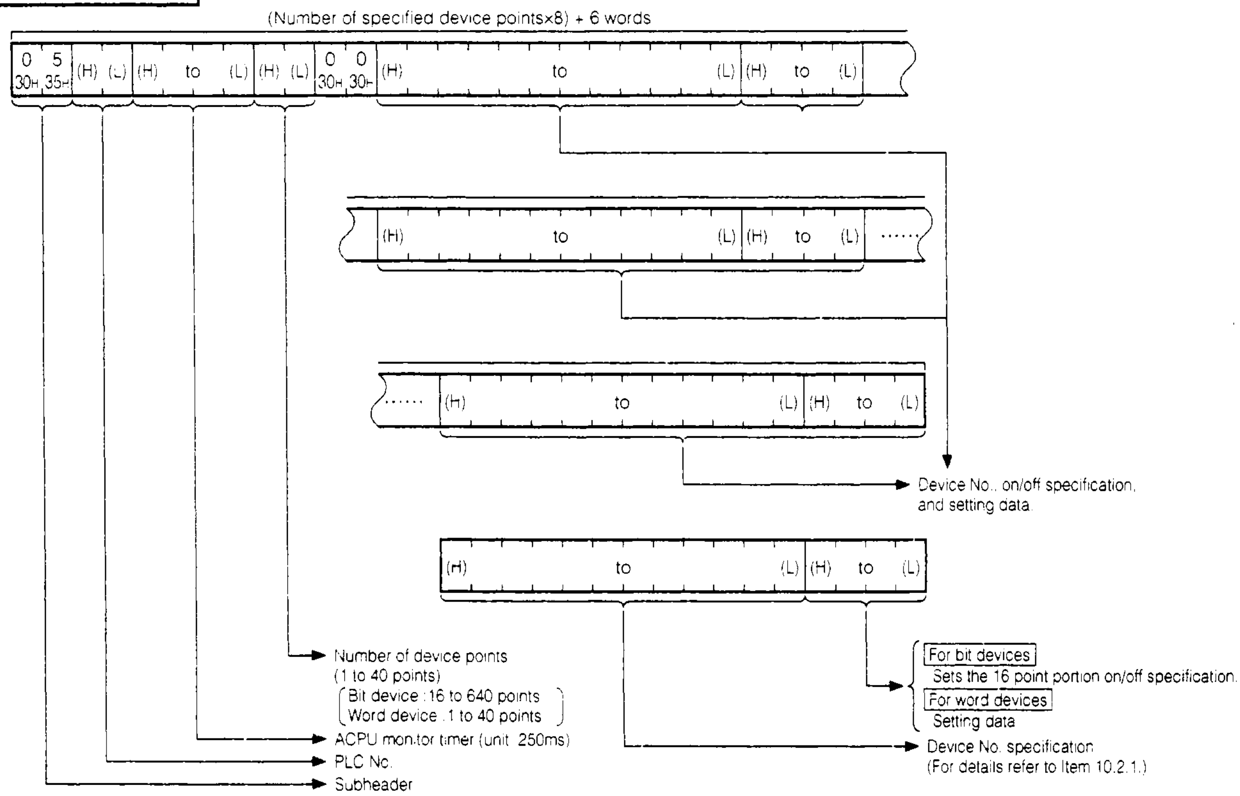


Response (E71 → remote node)

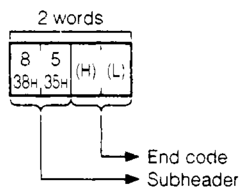


2 When exchanging using ASCII code

Command format



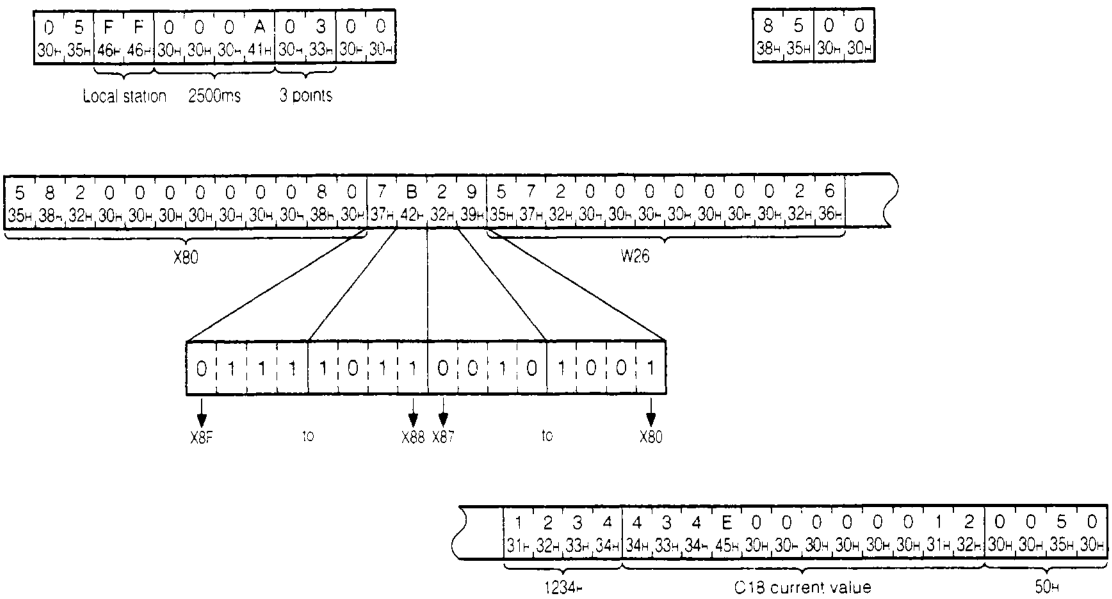
Response format



Example: When X80 to 8F is turned on/off and W26 is overwritten to "1234_H" and C18's current value is overwritten to "50_H" for the PLC CPU installed in E71.

Command (remote node → E71)

Response (E71 → remote node)



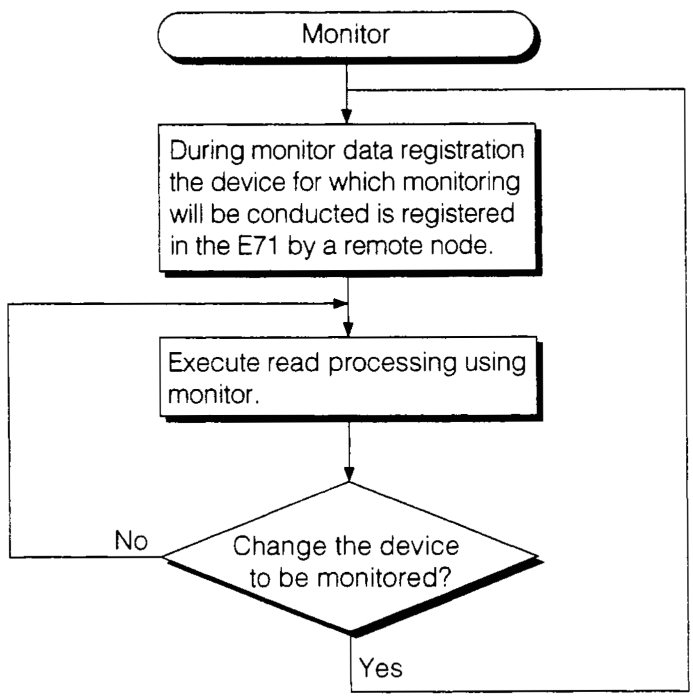
10.2.8 Device Memory Monitor

The device No. (device No. registered in the E71) on/off status and the contents can be monitored by a remote node by registering beforehand the device and device No. that you want to monitor with a remote node in the E71 and then executing a monitor instruction from the remote node.

Reading using device memory batch read can be processed in continuous device No., but by reading using the monitor it is possible to randomly specify a free device and No. and conduct the reading.

1 Monitor operation procedure

The operation procedure when conducting monitor is shown below.



Point

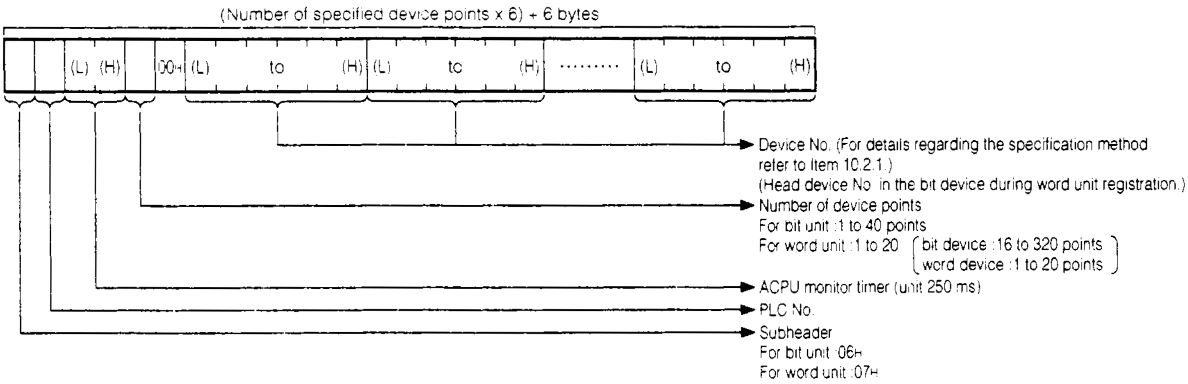
(1) In operation procedures like that above where monitoring will be executed, the monitor data registration operation must be conducted. If monitoring is executed without conducting monitor data registration, an error (End code 57_H) will occur.
(2) The monitor data registration contents will be erased if the power is turned off or the PLC CPU is reset.
(3) The 3 types of monitor data registration, device memory bit unit, word unit, and extension file register can be registered in the E71.
(4) When monitor data registration is performed from multiple remote nodes to the device memory of the PLC CPU on the same station, the registration data will be overwritten. Thus, the device memory last registered will become effective.

2 Monitor data registration

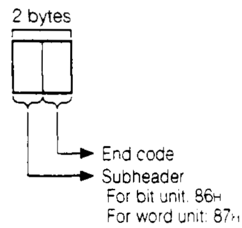
This section explains the command/response format when registering devices to be monitored.

(a) When exchanging using binary code

Command format



Response format



Example 1: Bit unit monitor data registration

When setting Y46, M12, and B2C in the PLC CPU installed in the E71.

Command (remote node → E71)

06H	FFH	0AH	00H	03H	00H	46H	00H	00H	00H	20H	59H	0CH	00H	00H	00H	20H	4DH	2CH	00H	00H	00H	20H	42H
	Local station	2500ms		3 points		Y46						M12						B2C					

Response (E71 → remote node)

86H	00H

Example 2: Word unit monitor data registration

When setting Y50 to 5F, D38, and W1E in the PLC CPU installed in the E71.

Command (remote node → E71)

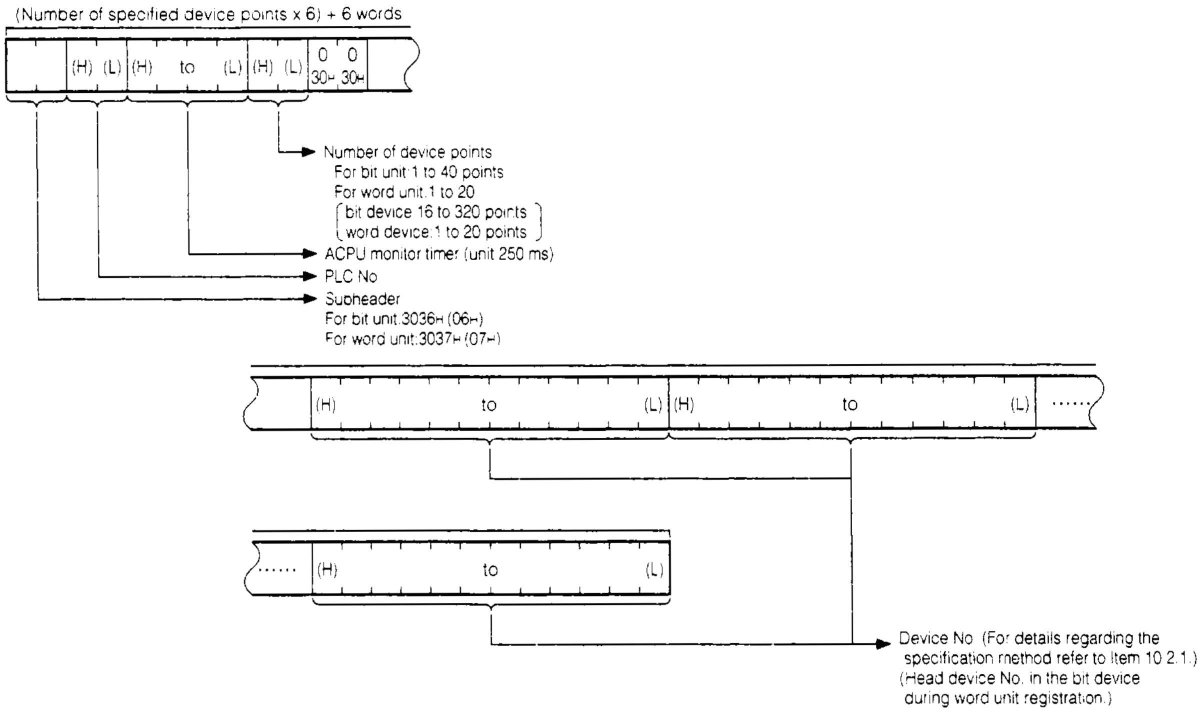
07H	FFH	0AH	00H	03H	00H	50H	00H	00H	00H	20H	59H	26H	00H	00H	00H	20H	44H	1EH	00H	00H	00H	20H	57H
	Local station	2500ms		3 points		Y50 (Set Y50 to 5F)						D38						W1E					

Response (E71 → remote node)

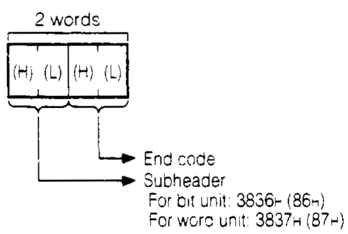
87H	00H

(b) When exchanging using ASCII code

Command format



Response format



Example 1: Bit unit monitor data registration

When setting Y46, M12, and B2C in the PLC CPU installed in the E71.

Command (remote node → E71)

0	6	F	F	0	0	0	A	0	3	0	0	5	9	2	0	0	0	0	0	0	0	4	6
30H	36H	46H	46H	30H	30H	30H	41H	30H	33H	30H	30H	35H	39H	32H	30H	00H	00H	00H	00H	00H	00H	34H	36H
		Local station		2500ms				3 points (03H)				Y46											

4	D	2	0	0	0	0	0	0	0	0	C	4	2	2	0	0	0	0	0	0	0	2	0
34H	44H	32H	30H	30H	30H	30H	30H	30H	30H	30H	43H	34H	32H	32H	30H	30H	30H	30H	30H	30H	30H	32H	43H
M12												B2C											

Response (E71 → remote node)

8	6	0	0
38H	36H	30H	30H

Example 2: Word unit monitor data registration

When setting Y50 to 5F, D38, and W1E in the PLC CPU installed in the E71.

Command (remote node → E71)

0	7	F	F	0	0	0	A	0	3	0	0	5	9	2	0	0	0	0	0	0	0	5	0
30H	37H	46H	46H	30H	30H	30H	41H	30H	33H	30H	30H	35H	39H	32H	30H	30H	30H	30H	30H	30H	30H	35H	30H
		Local station		2500ms				3 points (03H)				Y50											

4	4	2	0	0	0	0	0	0	0	2	6	5	7	2	0	0	0	0	0	0	0	1	E
34H	34H	32H	30H	30H	30H	30H	30H	30H	30H	32H	36H	35H	37H	32H	30H	30H	30H	30H	30H	30H	30H	31H	45H
D38												W1E											

Response (E71 → remote node)

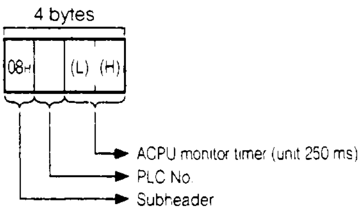
8	7	0	0
38H	37H	30H	30H

3 Bit unit monitor

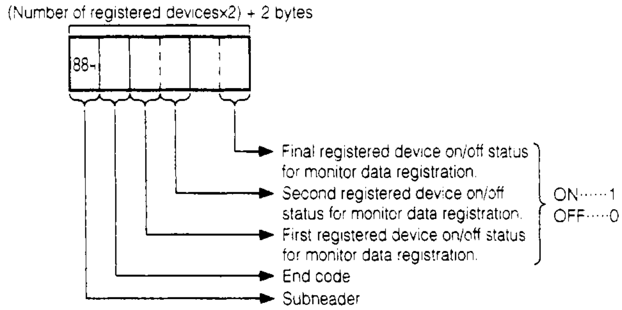
The following section explains the command/response format when conducting monitoring of a set bit device which monitor data registration has been conducted.

(a) When exchanging using binary code

Command format

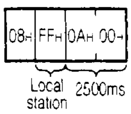


Response format

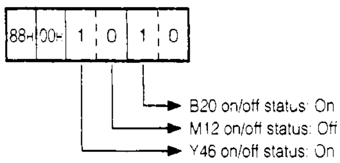


Example: When "Y46," "M12," and "B2C" are set using monitor data registration for the PLC CPU that is installed in the E71.

Command (remote node → E71)

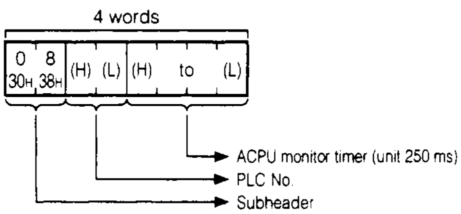


Response (E71 → remote node)

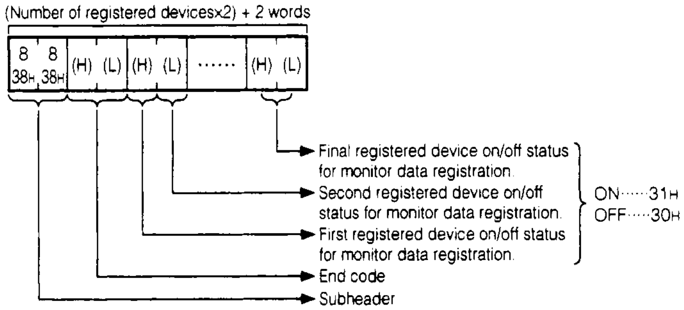


(b) When exchanging using ASCII code

Command format



Response format

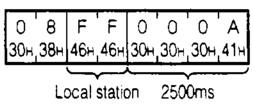


Remarks

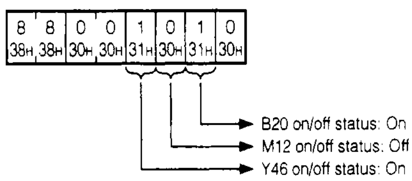
When the monitor registered number of devices is odd, dummy data (30_H) is added when monitor is executed. For example when the monitor registered number of devices is 3 points, the data portion returned is for 4 points. The last 1 byte is dummy data.

Example: When "Y46," "M12," and "B2C" are set using monitor data registration for the PLC CPU that is installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)

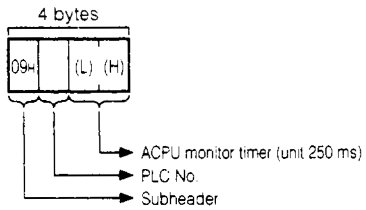


4 Word unit monitor

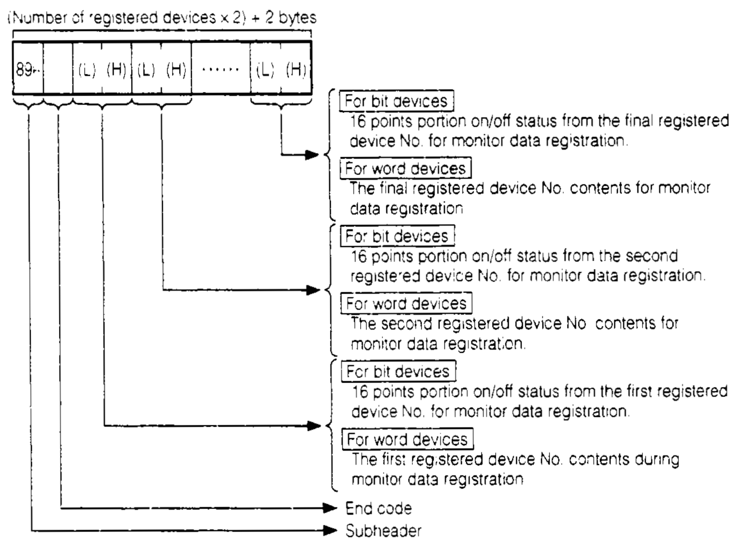
The following section explains the command/response format when conducting monitoring of a set word device and bit device (16 point unit) which monitor data registration has been conducted.

(a) When exchanging using binary code

Command format

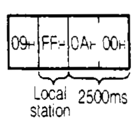


Response format

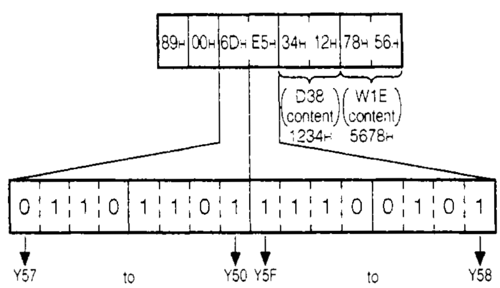


Example: When "Y50 to 5F," "D38," and "W1E" are set using monitor data registration for the PLC CPU that is installed in the E71.

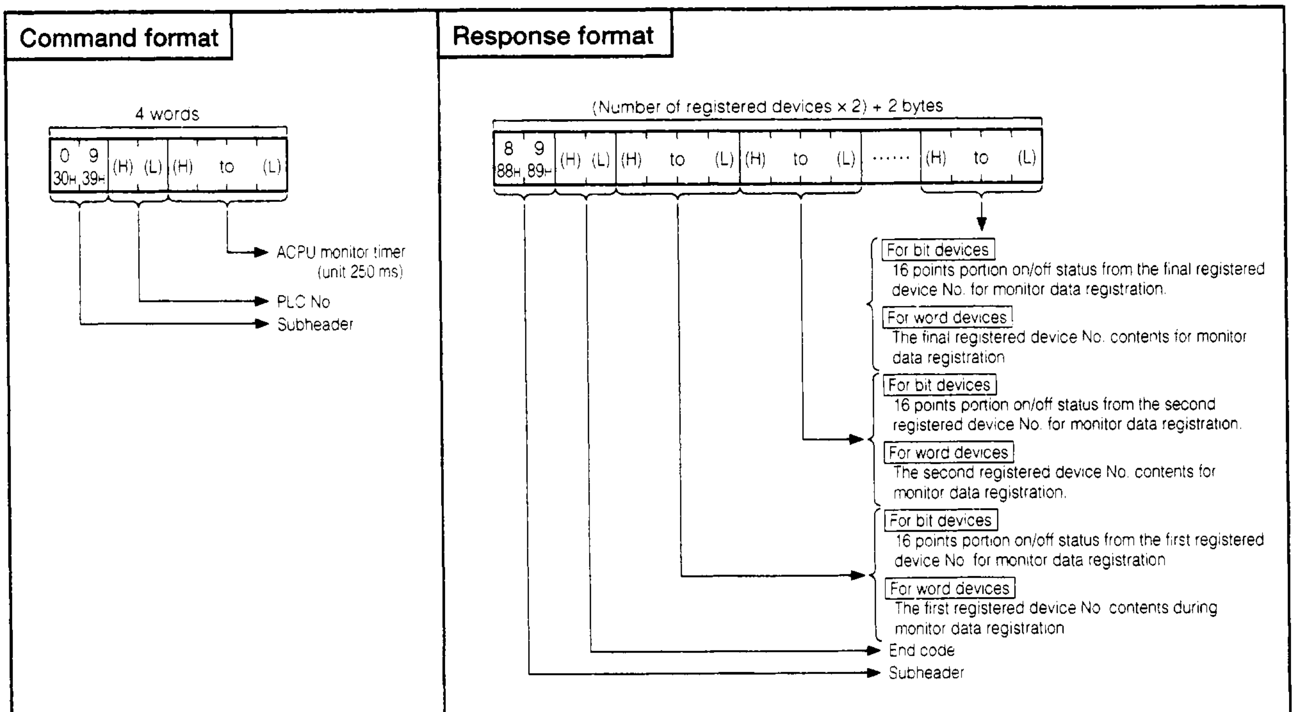
Command (remote node → E71)



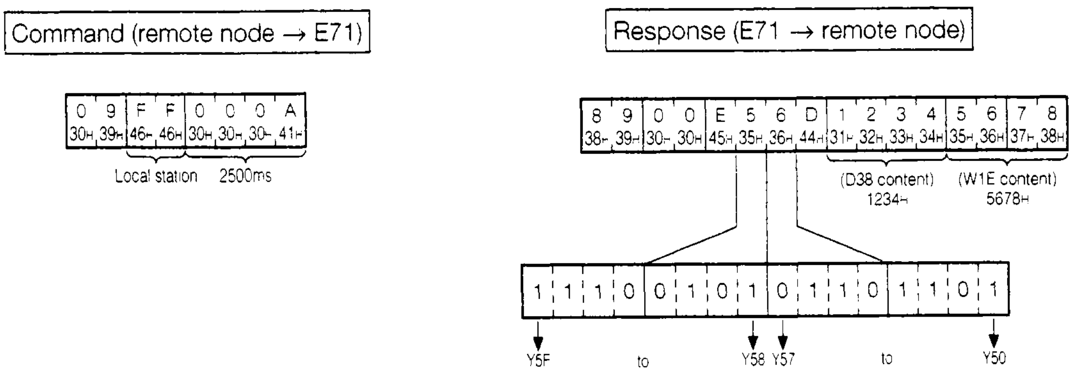
Response (E71 → remote node)



(b) When exchanging using ASCII code



Example: When "Y50 to 5F," "D38," and "W1E" are set using monitor data registration for the PLC CPU that is installed in the E71.



10.3 Extension File Register Read and Write

An extension file register is a file register that uses the empty area of the PLC CPU user memory area as a memory area for storing the data and calculated results required for the various types of data processing.

This section explains the control procedure specification contents, method, and an example specification for reading and writing the following extension file register.

10.3.1 Commands and Addresses

The function used for reading/writing extension file registers are shown in Table 10.3.

Table 10.3 Functions List

Item	Command/ response format	Processing description	Number of processing points conducted in one exchange	PLC CPU status		
				Stopped	Running	
					Write possible setting	Write impossible setting
Batch read	17_H	Extension file register (R) is read in 1 point unit.	256 points	O	O	O
Batch write	18_H	The extension file register (R) is written to in 1 point unit.	256 points	O	O	×
Test (random write)	19_H	The block No. and device No. are specified in the extension file register (R) is randomly written to in 1 point unit.	40 points	O	O	×
Monitor data registration	$1A_H$	The device No. to be monitored is registered in 1 point unit.	20 points	O	O	O
Monitor	$1B_H$	Monitor data registration is conducted and the extension file register is monitored.	———	O	O	O

In the PLC CPU status column in the above table the "O" represents execution possible and the "X" represents execution not possible.

Extension file register address

(a) Number of extension file registers.

Block No. 0 .... The number of registers specified by the PLC CPU parameter.

After block No. 1 .... There are 8192 registers for each block.

(b) The specifiable block No. range varies depending on the PLC CPU memory capacity (memory cassette type) and the PLC CPU parameter setting. For details refer to the SW□GHP-UTLPLC-FN1 Utility Package Operating Manual or the AnA/AnU Programming Manual (Dedicated Instruction Edition).

10.3.2 Precautions When Reading/Writing Extension File Registers

This section explains the precautions when reading/writing extension file registers.

(1) Extension file registers cannot use A1 and A1NCPU.

(2) Reading and writing cannot be performed for QnACPU extension file registers.

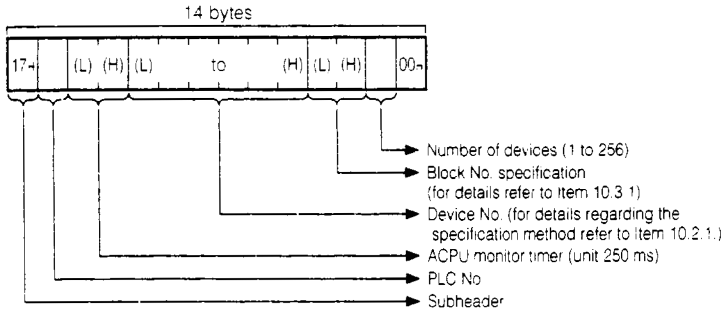
(3) An error (End code 58H) sometimes cannot be detected even when read/write is executed for a block No. that does not exist. In this case, the read data is not correct data. In addition, when write is conducted the PLC CPU's user memory can be corrupted.

10.3.3 Extension File Register Batch Read

This section explains the command/response format when executing an extension file register batch read.

1 **When exchanging using binary code**

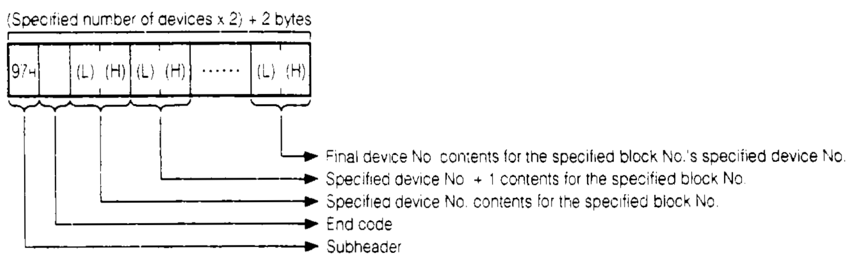
Command format



Remarks

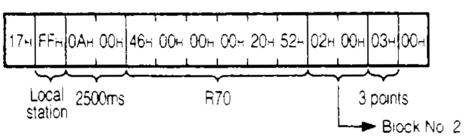
Set to "00H" when specifying the number of devices as 256.

Response format

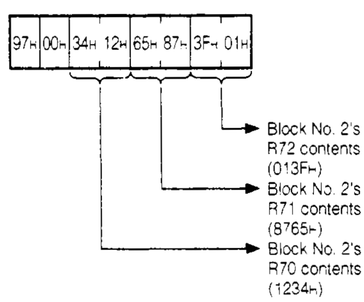


Example: When the contents of extension file register No. 2 block's R70 to 72 for the PLC CPU installed in the E71 are read.

Command (remote node → E71)

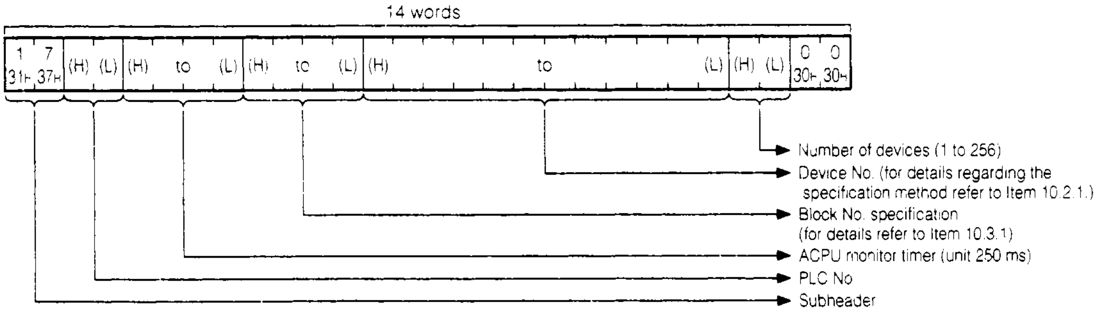


Response (E71 → remote node)



2 When exchanging using ASCII code

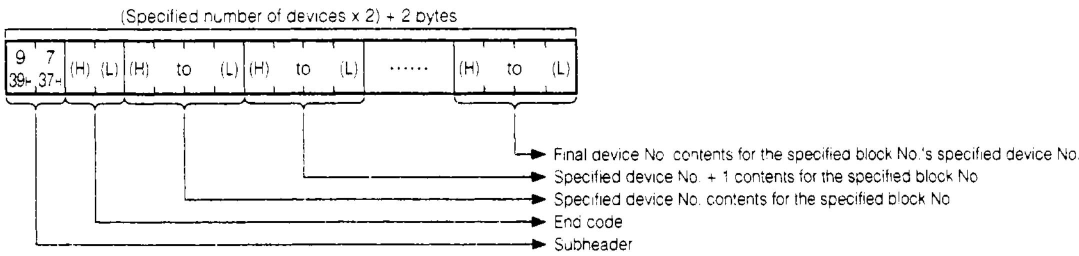
Command format



Remarks

Set to "00H" when specifying the number of devices as 256.

Response format



Example: When the contents of extension file register No. 2 block's R70 to 72 for the PLC CPU installed in the E71 are read.

Command (remote node → E71)

1 7	F F	0 0 0 A	0 0 0 2	5 2 2 0 0 0 0 0 0 0 4 6	0 3	0 0
31H 37H	46H 46H	30H 30H 30H 41H	30H 30H 30H 32H	35H 32H 32H 30H 30H 30H 30H 30H 30H 30H 34H 36H	30H 33H	30H 30H
	Local station	2500ms	Block No. 2	R70	3 points	

Response (E71 → remote node)

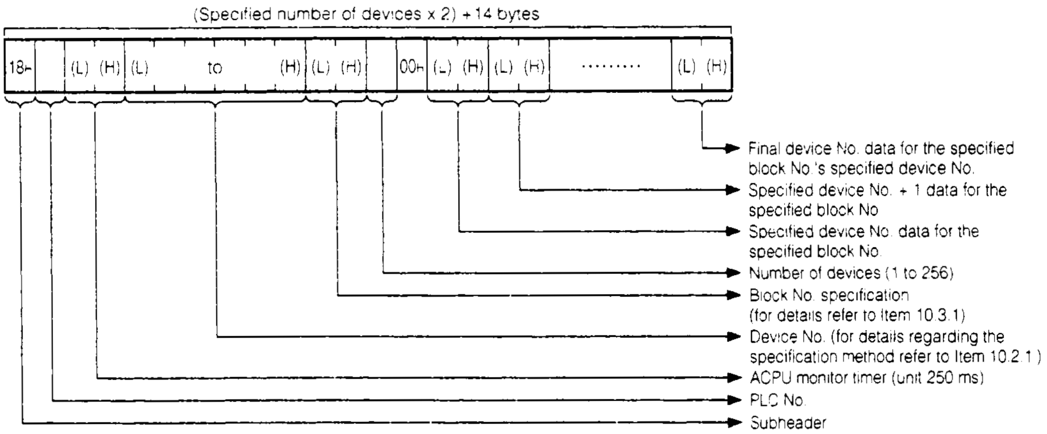
9 7	0 0	1 2 3 4	8 7 6 5	0 1 3 F
39H 37H	30H 30H	31H 32H 33H 34H	38H 37H 36H 35H	30H 31H 33H 46H
		Block No. 2's R70 contents (1234H)	Block No. 2's R71 contents (8765H)	Block No. 2's R72 contents (013FH)

10.3.4 Extension File Register Batch Write

This section explains the command/response format when executing an extension file register batch write.

1 When exchanging using binary code

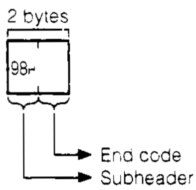
Command format



Remarks

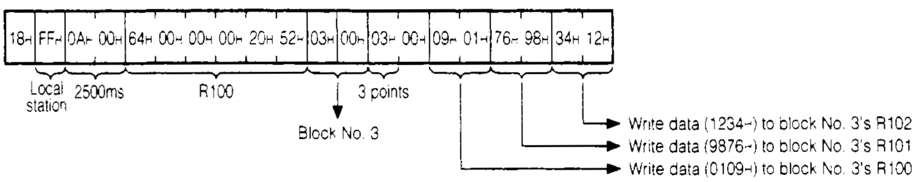
Set to "00H" when specifying the number of devices as 256.

Response format

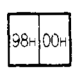


Example: When the contents of extension file register No. 3 block's R100 to 102 for the PLC CPU installed in the E71 are written.

Command (remote node → E71)

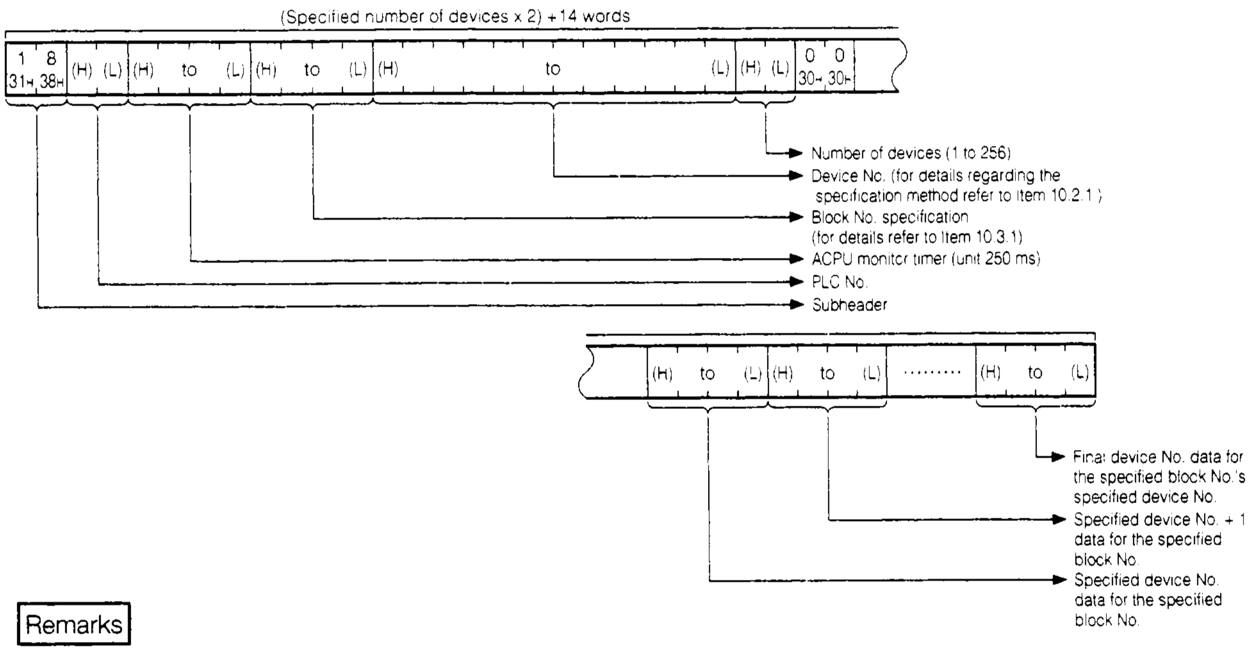


Response (E71 → remote node)



2 When exchanging using ASCII code

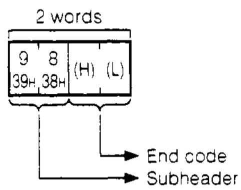
Command format



Remarks

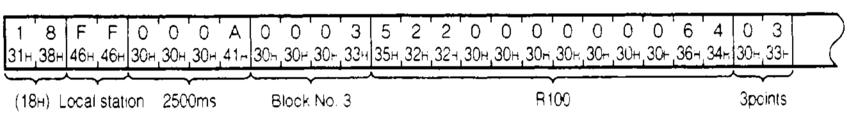
Set to "3030H" when specifying the number of devices as 256.

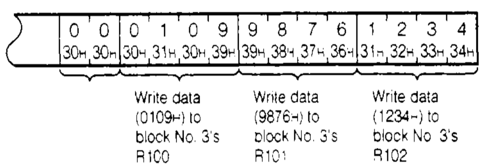
Response format



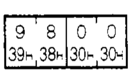
Example: When the contents of extension file register No. 3 block's R100 to 102 for the PLC CPU installed in the E71 are written.

Command (remote node → E71)





Response (E71 → remote node)

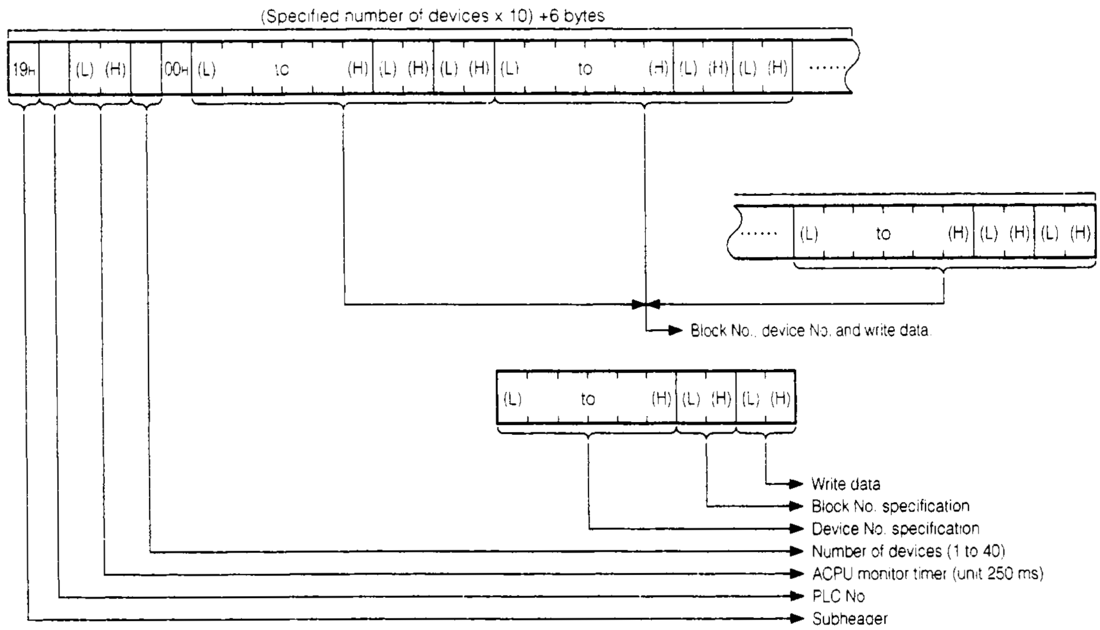


10.3.5 Extension File Register Test (Random Write)

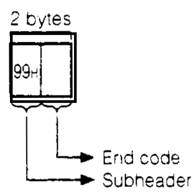
This section explains the command/response format when executing an extension file register random write.

1 When exchanging using binary code

Command format



Response format



Example: When the contents of extension file register No. 2 block's R26 and No.3 Block's R19 for the PLC CPU installed in the E71 are written.

Command (remote node → E71)

19H	FFH	0AH	00H	02H	00H	1AH	00H	00H	00H	20H	52H	02H	00H	34H	12H	13H	00H	00H	00H	20H	52H	03H	00H	09H	01H

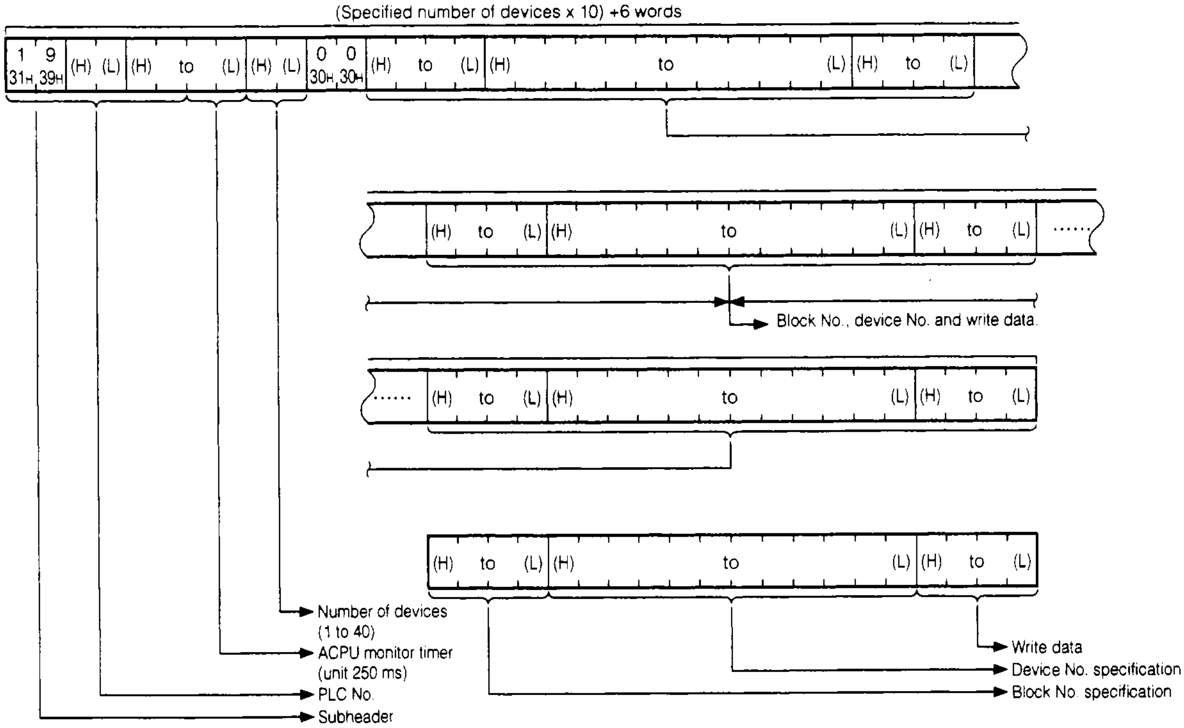
Local station; 2500ms; 2 points; Write data (1234H) to block No. 2's R26; Write data (0109H) to block No. 3's R19

Response (E71 → remote node)

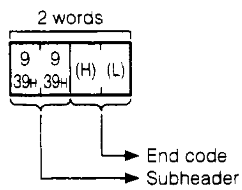
99H	00H

2 When exchanging using ASCII code

Command format

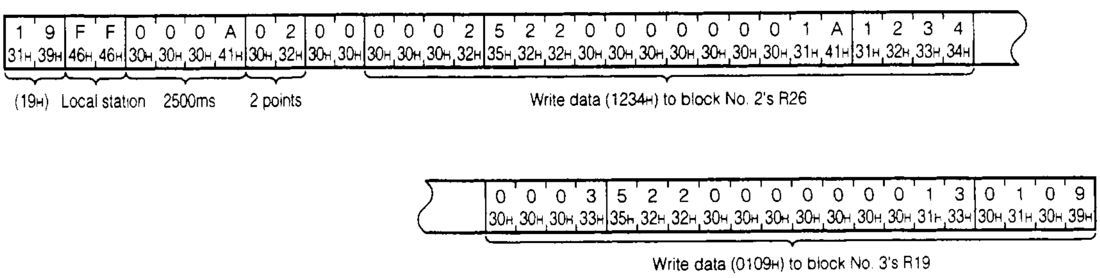


Response format

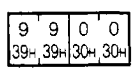


Example: When the contents of extension file register No. 2 block's R26 and No.3 Block's R19 for the PLC CPU installed in the E71 are written.

Command (remote node → E71)



Response (E71 → remote node)



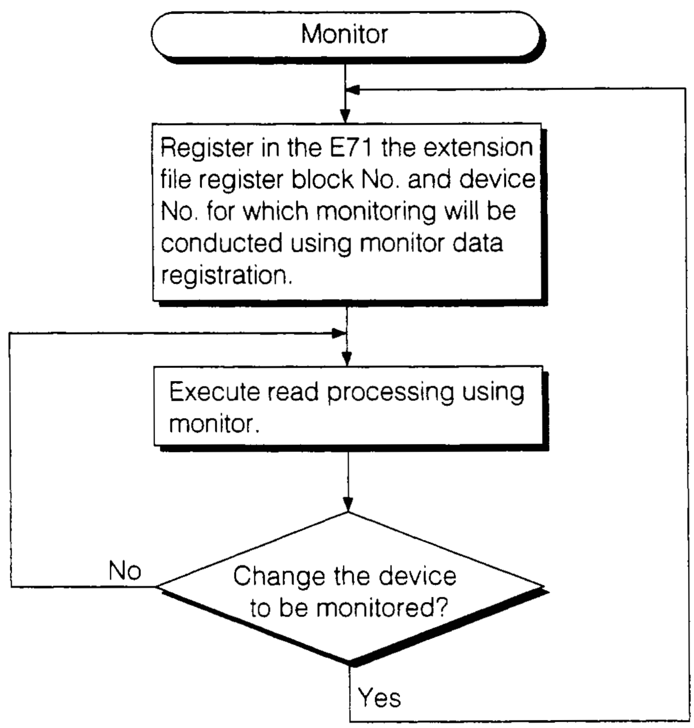
10.3.6 Extension File Register Monitor

The extension file register in the PLC CPU (device No. registered in the E71) the contents can be monitored by a remote node by registering beforehand the extension file register block No. and device No. that you want to monitor with a remote node in the E71 and then executing a monitor instruction from the remote node.

Reading using extension file register batch read can be processed in continuous device No., but by reading using the monitor it is possible to randomly specify a free device and No. and conduct the reading.

1 Monitor operation procedure

The operation procedure when conducting monitor is shown below.



Point

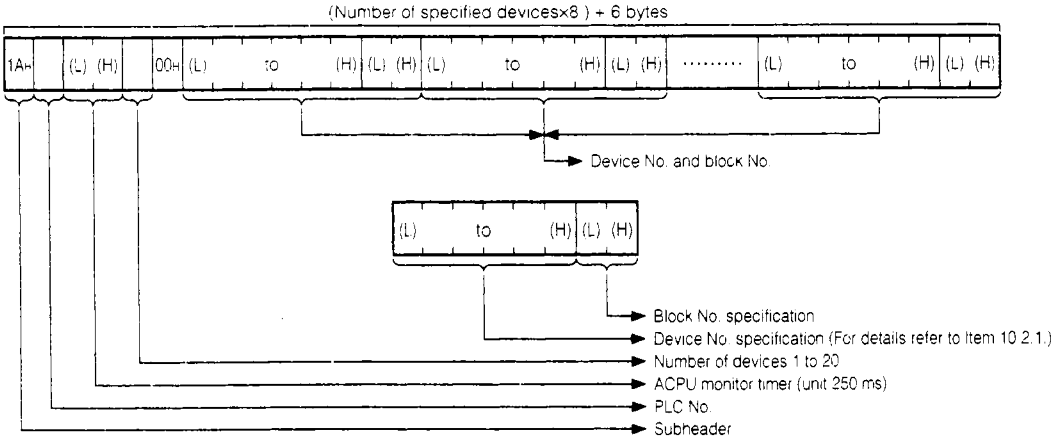
(1) In operation procedures like that above where monitoring will be executed, the monitor data registration operation must be conducted. If monitoring is executed without conducting monitor data registration, an error (End code 57_H) will occur.
(2) The monitor data registration contents will be erased if the power is turned off or the PLC CPU is reset.
(3) The 3 types of monitor data registration, device memory bit unit, word unit, and extension file register can be registered in the E71.
(4) When monitor data registration is performed from multiple remote nodes to the device memory of the PLC CPU on the same station, the registration data will be overwritten. Thus, the device memory last registered will become effective.

2 Monitor data registration

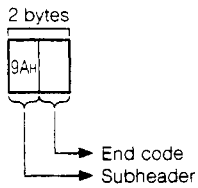
This section explains the command/response format when registering the extension file register device to be monitored.

(a) When exchanging using binary code

Command format

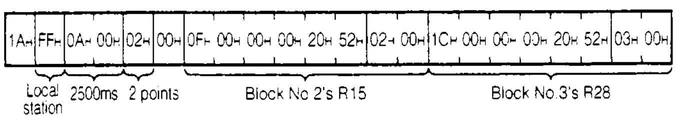


Response format

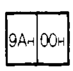


Example: When setting the extension file register No. 2 block's R15 and No. 3 block's R28 of the PLC CPU installed in the E71.

Command (remote node → E71)

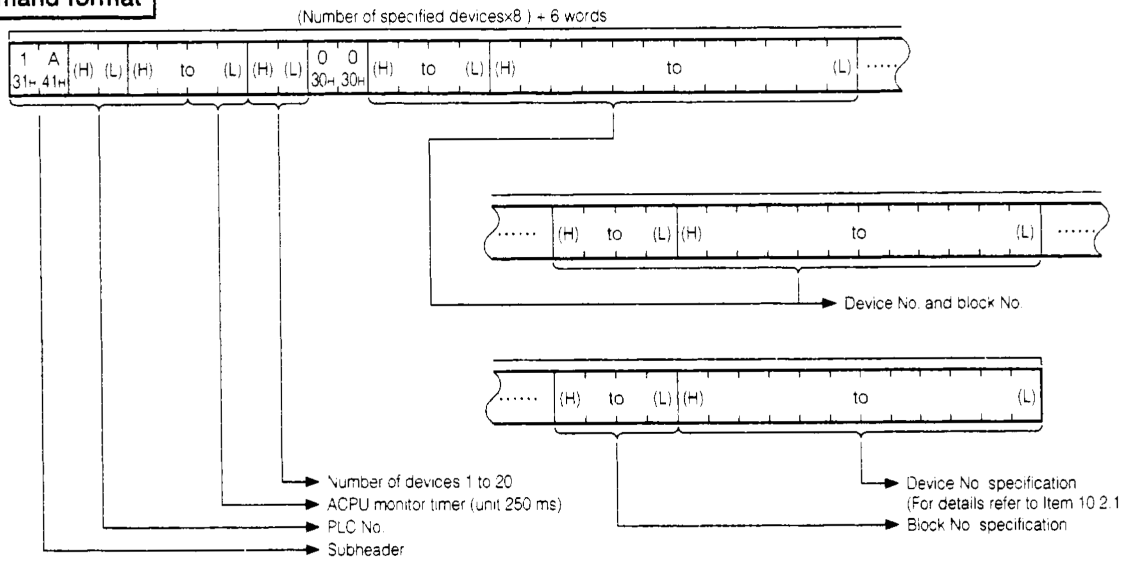


Response (E71 → remote node)

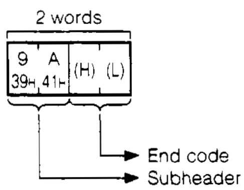


(b) When exchanging using ASCII code

Command format

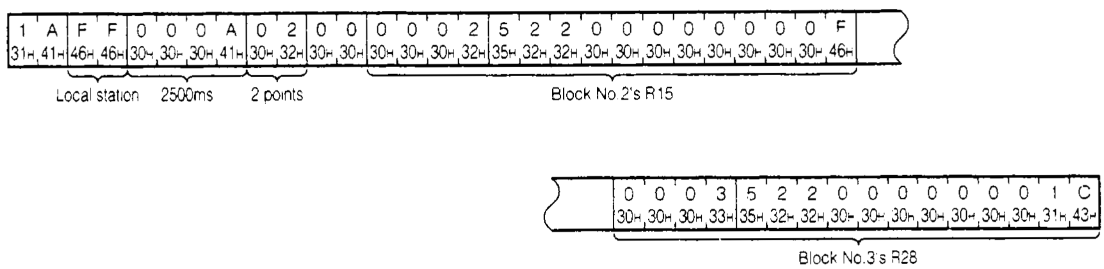


Response format

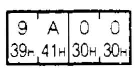


Example: When setting the extension file register No. 2 block's R15 and No. 3 block's R28 of the PLC CPU installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)

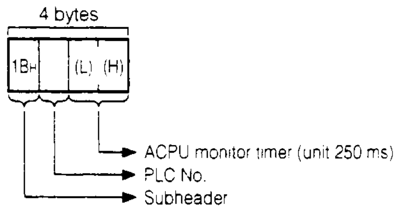


3 Monitor

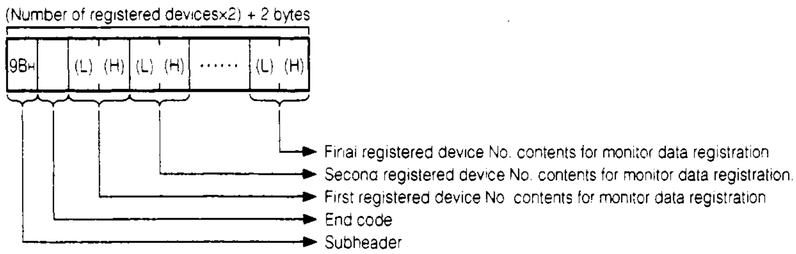
The following section explains the command/response format when conducting monitoring of a set extension file register which monitor data registration has been conducted.

(a) When exchanging using binary code

Command format

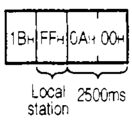


Response format

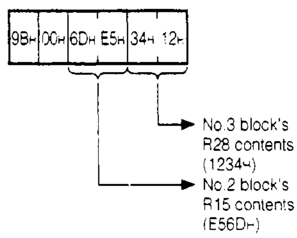


Example: When No.2 block's R15 and No.3 block's R28 are set using monitor data registration for the PLC CPU that is installed in the E71.

Command (remote node → E71)

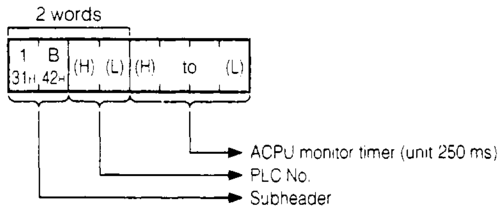


Response (E71 → remote node)

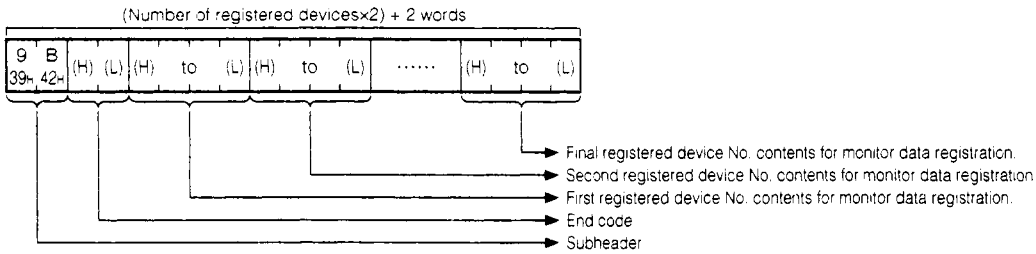


(b) When exchanging using ASCII code

Command format

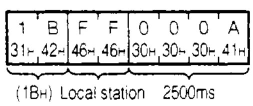


Response format

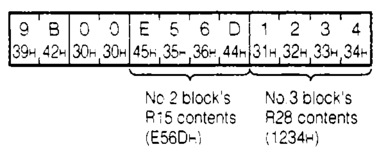


Example: When No.2 block's R15 and No.3 block's R28 are set using monitor data registration for the PLC CPU that is installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)



10.3.7 Extension File Register Direct Read/Write

1 **The AnACPU dedicated commands used for extension file register direct read/write are shown below. These command functions are used to access the extension file registers in block No. 0 to No. 256, and can specify the addresses from the block No. 1 device No. 0 as device No. without concern for each block No. and can access them. (The usable number of blocks × 8192 extension file registers can be accessed using continuous device Nos.)**

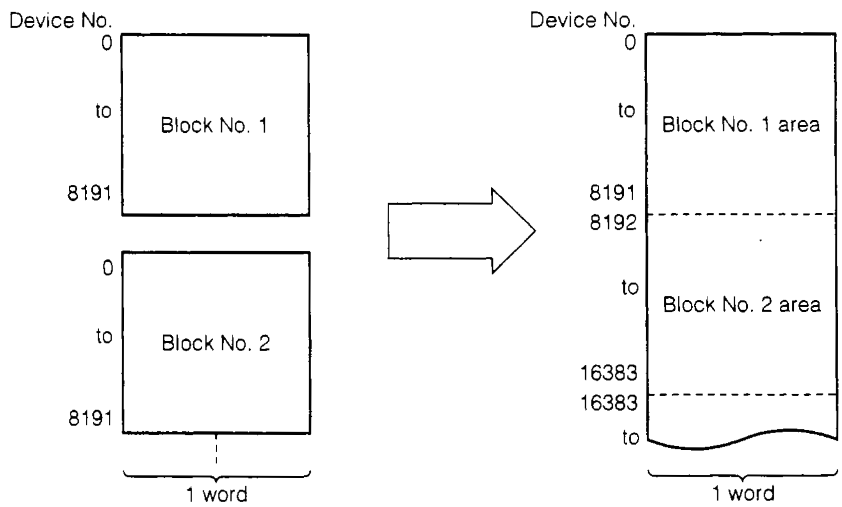
Item	Command/ response format	Processing description	Number of processing points conducted in one exchange	PLC CPU status		
				Stopped	Running: Write possible setting	Running: Write impossible setting
Direct read	$3B_H$	Reads the extension file register (R) in one point unit.	256 points	○	○	○
Direct write	$3C_H$	Writes the extension file register (R) in one point unit.	256 points	○	○	×

In the PLC CPU status column in the above table the "○" stands for executing possible and the "×" stands for execution not possible.

2 **Extension file register device No.**

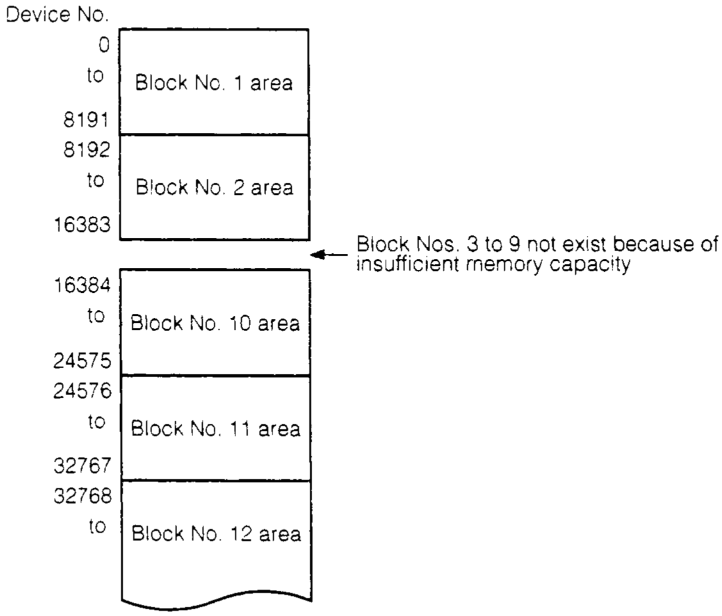
(a) The device No. range that can be specified is shown below.

0 to (usable number of blocks x 8192) – 1



The device Nos. that will be used for direct read/write are automatically allocated in order from the smallest device of the block No. from those after block No. 1. The device Nos. that can be specified vary depending on the type of memory cassette and PLC CPU parameter setting.

Device Nos. are not allocated for block Nos. not existing in the memory cassette. However, device Nos. are automatically allocated skipping the block Nos. that do not exist in the memory cassette.



Point

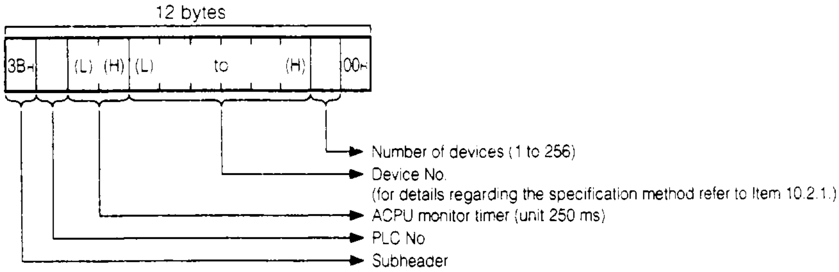
(1) The AnACPU dedicated commands can only be used when executing read or write of data in the extension file registers for blocks Nos. 0 to 256. In addition, these commands can be used regardless of whether the parameter file register settings are valid.
(2) When accessing the specified file register (R) using parameters or when accessing by specifying the block No., use the commands given in Item 10.3.6.
(3) The calculation method for the header device No. specified using the AnACPU dedicated commands is as follows.
Given that the device No. of the nth block from the header is m(0 to 8191), then
Header device No. = (n – 1) x 8192 + m

3 Extension file register direct read

This section explains the command/response format when executing an extension file register direct read.

(a) When exchanging using binary code

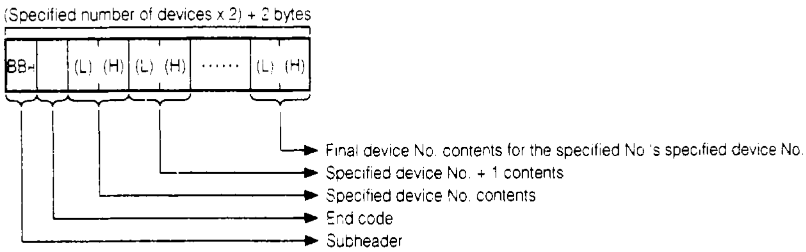
Command format



Remarks

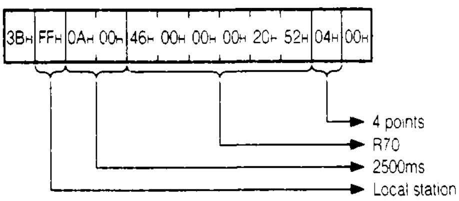
Set to "00H" when specifying the number of devices as 256.

Response format

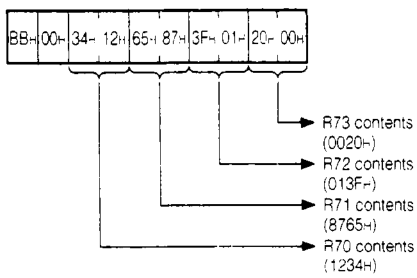


Example: When the contents of extension file register R70 to 73 for the PLC CPU installed in the E71 are read.

Command (remote node → E71)

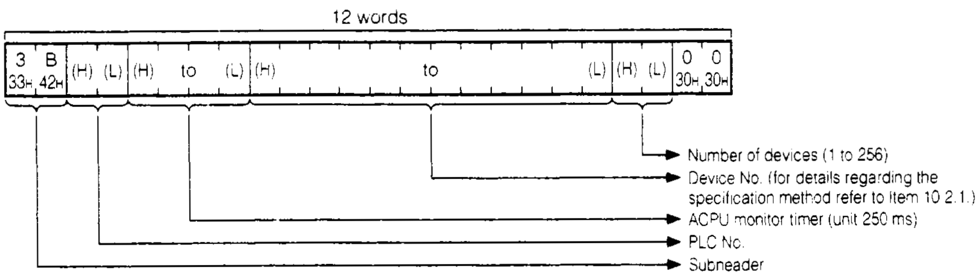


Response (E71 → remote node)



(b) When exchanging using binary code

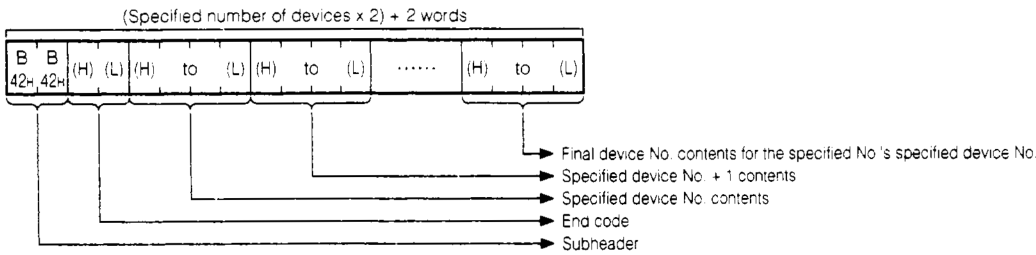
Command format



Remarks

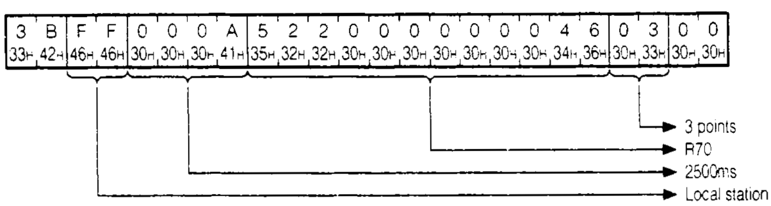
Set to "00H" when specifying the number of devices as 256.

Response format

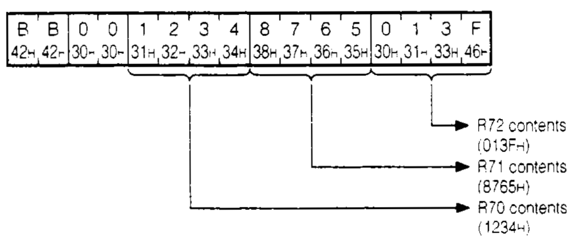


Example: When the contents of extension file register R70 to 72 for the PLC CPU installed in the E71 are read.

Command (remote node → E71)



Response (E71 → remote node)

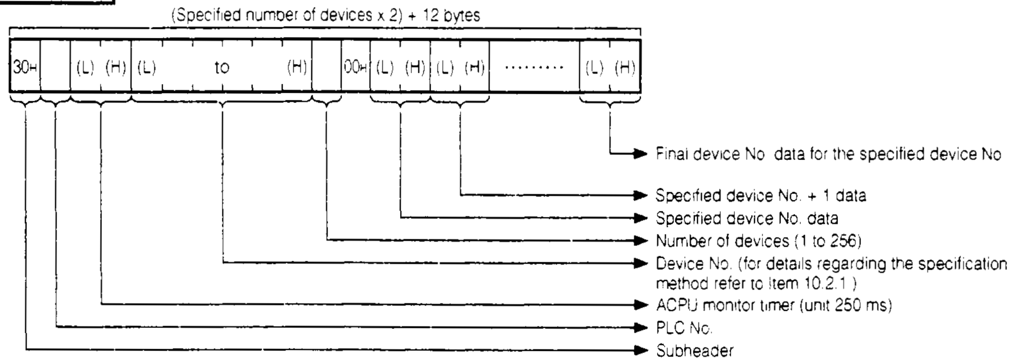


4 Extension file register direct write

This section explains the command/response format when executing an extension file register direct write.

(a) When exchanging using binary code

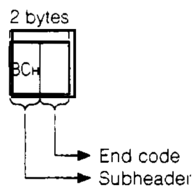
Command format



Remarks

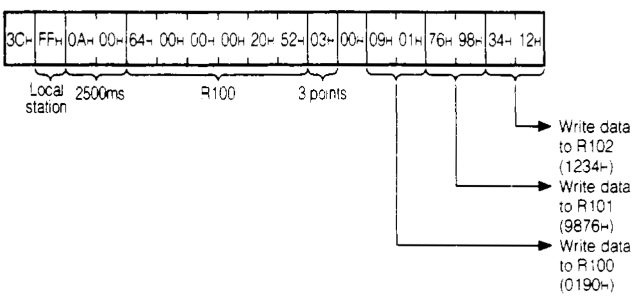
Set to "00_H" when specifying the number of devices as 256.

Response format

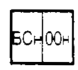


Example: When the contents of extension file register R100 to 102 data are written in the PLC CPU installed in the E71 are read.

Command (remote node → E71)

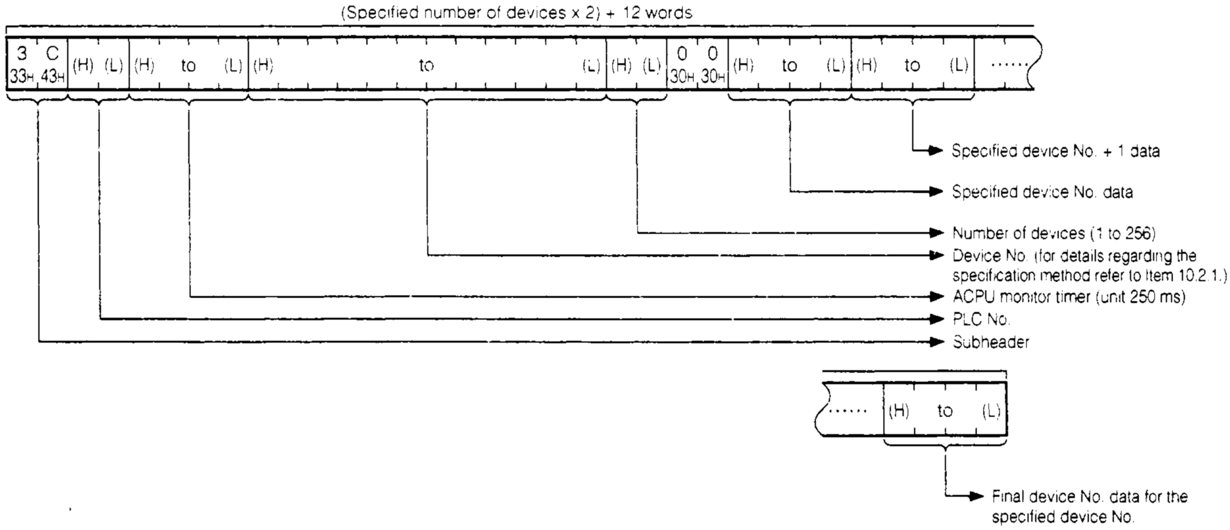


Response (E71 → remote node)



(b) When exchanging using ASCII code

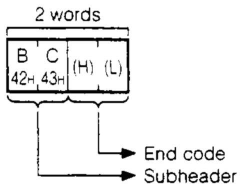
Command format



Remarks

Set to "3030H" when specifying the number of devices as 256.

Response format



Example: When the contents of extension file register R100 to 102 data are written in the PLC CPU installed in the E71 are read.

Command (remote node → E71)

3C	FF	000A	5220000000000064	03	00	0109	9876	1234
33H,43H	46H,46H	30H,30H,30H,41H	35H,32H,32H,30H,30H,30H,30H,30H,30H,30H,36H,34H	30H,30H	30H,30H	30H,31H,30H,39H	39H,38H,37H,36H	31H,32H,33H,34H
	Local station	2500ms	R100	3 points		Write data to R100 (0109H)	Write data to R101 (9876H)	Write data to R102 (1234H)

Response (E71 → remote node)

BC	00
42H,43H	30H,30H

10.4 Special Function Module Data Read and Write

This section explains the control procedure specification contents, method, and example specification when reading contents from the special function module buffer memory area or writing data to this buffer memory area.

This command accesses the special function module buffer memory in byte units.

10.4.1 Command and Data Item Specification Method

(1) The functions used to read from and write to the special function module are shown in Table 10.4.

Table 10.4 Functions List

Item	Command/ response classification	Processing description	Number of processing points conducted in one exchange	PLC CPU status		
				Stopped	Running	
					Write possible setting	Write impossible setting
Batch read	0EH	Reads the special function module buffer memory contents.	256 bytes (128 words)	○	○	○
Batch write	0FH	Writes data to the special function module buffer memory.		○	○	×

In the PLC CPU status column in the above table the "○" means execute possible and the "×" means execute not possible.

(2) Link possible special function module model name, buffer memory head address, and module No.

Special function module model name	Buffer memory head address (hexadecimal)	Module No. installed in slot 0
Model AD61 (S1) high speed counter module	80H	01H
Model A616AD analog-digital conversion module	10H	01H
Model A616DAI digital-analog conversion module	10H	01H
Model A616DAV digital-analog conversion module	10H	01H
Model A616TD temperature-digital conversion module	10H	01H
Model A62DA (S1) digital-analog conversion module	10H	01H
Model A68AD (S2) analog-digital conversion module	80H	01H
Model A68ADN analog digital conversion module	80H	01H
Model A68DAV/DAI (S1) digital-analog conversion module	10H	01H
Model A68RD3/4 temperature-digital conversion module	10H	01H
Model A84AD analog-digital conversion module	10H	02H
Model A81CPU PID control module	200H	03H
Model A61LS position detection module	80H	01H
Model A62LS (S5) position detection module	80H	02H
Model AJ71 (P) T32 (S3) MELSECNET/MINI master module	20H	01H
Model AJ61BT11 CC-Link system master·local module	2000H (*2)	01H
Model AJ71C22 (S1) multiple drop link module	1000H	01H
Model AJ71C24 (S3/S6/S8) computer link module	1000H	01H
Model AJ71UC24 computer link module	400H	01H
Model AD51 (S3) intelligent communication module	800H	02H
Model AD51H (S3) intelligent communication module	800H	02H

Special function module model name	Buffer memory head address (hexadecimal)	Module No. installed in slot 0
Model AJ71C21 (S1) terminal interface module	400H	01H
Model AJ71B62 (S3) B/NET interface module	20H	01H
Model AJ71P41 SUMINET interface module	400H	01H
Model AJ71E71 (S3) Ethernet interface module	400H (*3)	01H
Model AD51FD (S3) external problem diagnostic module	280H	02H
Model AD57G (S3) graphic controller module	280H	02H
Model AS25VS vision sensor module	100H	02H
Model AS50VS vision sensor module	100H	02H
Model AS50VS-GN vision sensor module	80H	02H
Model AD59 (S1) memory card interface module	1800H (*1)	01H
Model AJ71ID1(2)-R4 ID interface module	280H	01H
Model AD70 (D) (S2) positioning module	80H	01H
Model AD71 (S1/S2/S7) positioning module	200H	01H
Model AD72 positioning module	200H	02H
Model AD75P1/P2/P3 (S3), AD75M1/M2/M3 positioning module	800H	01H
Model A1SD61, A1SD62 (E/D (S1)) high speed counter module	10H	01H
Model A1S62DA digital-analog conversion module	10H	01H
Model A1S62RD3/4 temperature-digital conversion module	10H	01H
Model A1S64AD analog-digital conversion module	10H	01H
Model A1SJ71 (U) C24-R2 (R4/PRF) computer link module	400H	01H
Model A1SJ71E71-B2/B5 (S3) Ethernet interface module	400H (*3)	01H
Model A1SD51S intelligent communication module	800H	01H
Model A1SJ71ID1(2)-R4 ID interface module	280H	01H
Model A1SD70 single axis positioning module	80H	01H
Model A1SD71-S2 (S7) positioning module	200H	02H
Model A1SD75P1/P2/P3 (S3) positioning module	800H	01H
Model A1SD75M1/M2/M3 positioning module	800H	01H
Model A1S63ADA analog I/O module	10H	01H
Model A1S64TCTT (BW)-S1 temperature adjustment module	20H	01H
Model A1S64TCRT (BW)-S1 temperature adjustment module	20H	01H
Model A1S62TCTT (BW)-S2 temperature adjustment module	20H	01H
Model A1S62TCRT (BW)-S2 temperature adjustment module	20H	01H
Model A1S68DAV/DAI digital-analog conversion module	20H	01H
Model A1S68AD analog-digital conversion module	20H	01H
Model A1S68TD temperature-digital conversion module	20H	01H
Model A1SJ71PT32-S3 MELSECNET/MINI master module	20H	01H
Model A1SJ61BT11 CC-Link system master·local module	2000H (*2)	01H

*1 Only the memory area for memory card access can be read from/written to when the memory card bank is switched by the I/O signal Y10, Y11 between the PLC CPU and AD59 (S1).

*2 By switching banks of the buffer memory by the input/output signal Y1C/Y1D between the PLC CPU and AJ61BT11/A1SJ61BT11, the buffer memory of the corresponding bank can be read/written.

*3 By switching banks of the buffer memory by the input/output signal Y1C between the PLC CPU and E71, the fixed buffer of the corresponding bank and the buffer for random access can be read/written.

3 Thinking regarding a special function module buffer memory

Special function module buffer memory is configured of 1 address of 16 bits (1 word) and is read from and written to using a FROM/TO instruction between the PLC CPU and the special function module. When the special function module buffer memory is read from or written to from a remote node via the E71, the read/write is conducted using 1 address = 8 bits (1 byte) units.

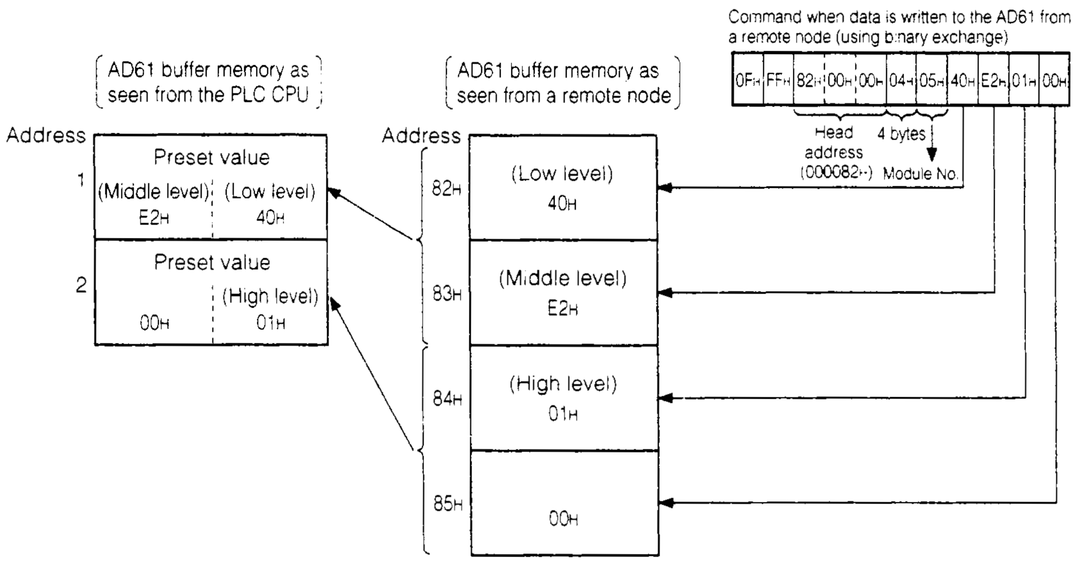
The address (hexadecimal) specified by a remote node is calculated using the FROM/TO instruction address as shown below.

Specified address (hexadecimal) = [(FROM/TO instruction address x 2)] is made hexadecimal + each module head address

Example: When the model AD61 high speed counter module FROM/TO instruction address 1 (preset value) is specified

Specified address		FROM/TO instruction address 1 x 2		Head address
82_H	=	2_H	+	80_H

Following is an explanation of an example using the AD61 for the data format when the special function module buffer memory is accessed from a remote node via the E71.



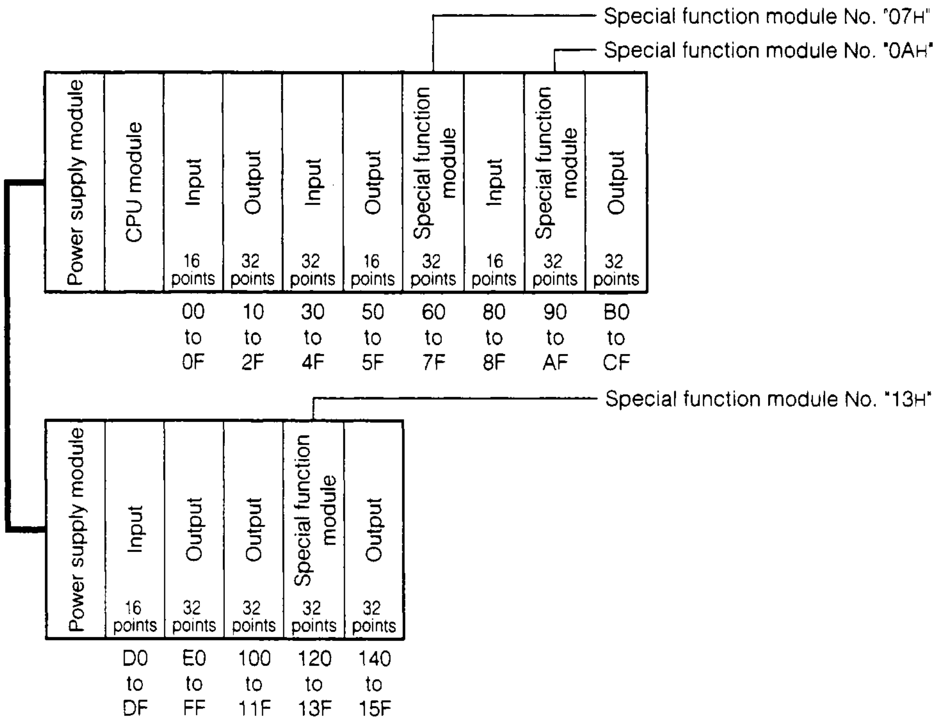
Point

The special function module buffer memory contains a read/write possible area, read only area, write only area, and an OS user usage not possible area for each module. Execute this function in accordance with the explanations given in each module manual. Conducting a mistake in read/write will cause an error to occur in the PLC CPU or the special function module.

4 **Thinking regarding special function module Nos. occurring in commands**

(a) Module Nos. of special function modules that occupy one slot

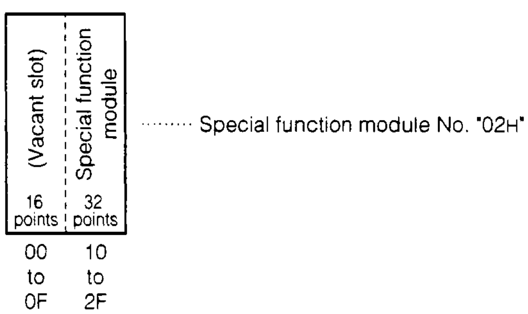
For special function module Nos. that are specified by control procedures, if the special function module I/O address final address is expressed in 3 digits, then only the first 2 digits are used.



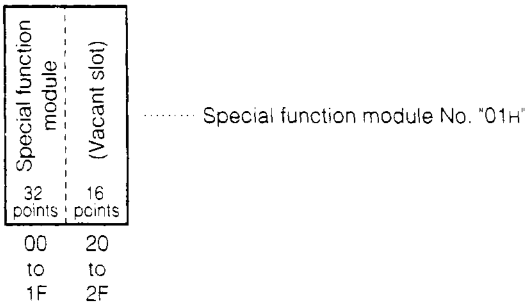
(b) Module Nos. of special function modules that occupy 2 slots

For special function modules that occupy 2 slots, the number of occupied points for each slot of each module is set. Special function module Nos. for which the final address of a slot allocated as a special function module is expressed in 3 digits, only the first 2 digits are used. For information regarding the allocation of each slot of each module, refer to the Special Function Module Users Manual.

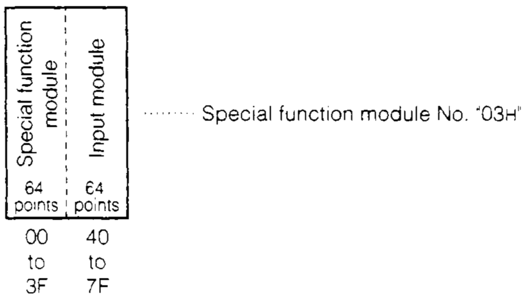
① Module when the first half of the slots are allocated as vacant slots. (AD72, A84AD, etc.)



② Module when the last half of the slots are allocated as vacant slots. (A61LS, etc.)



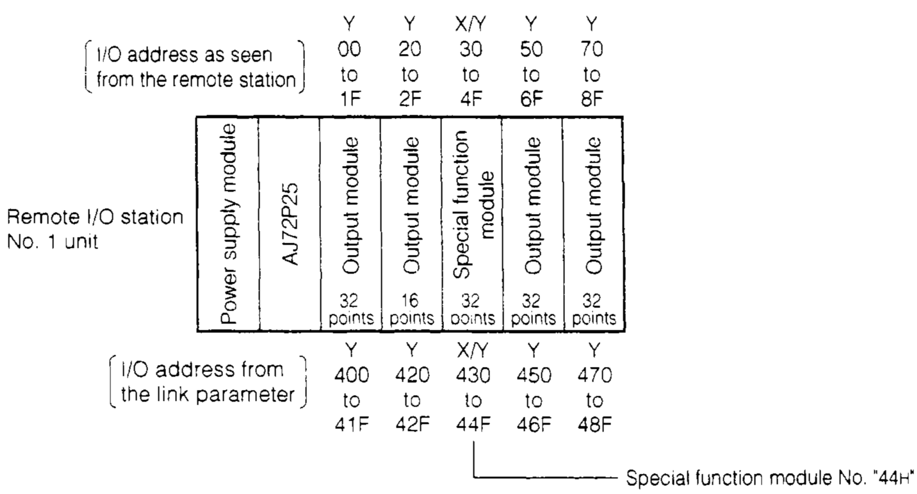
③ Module when the special function module allocation I/O allocation is mixed. (For the A81CPU)



(c) MELSECNET(II) and MELSECNET(B) remote station special function module's module No.

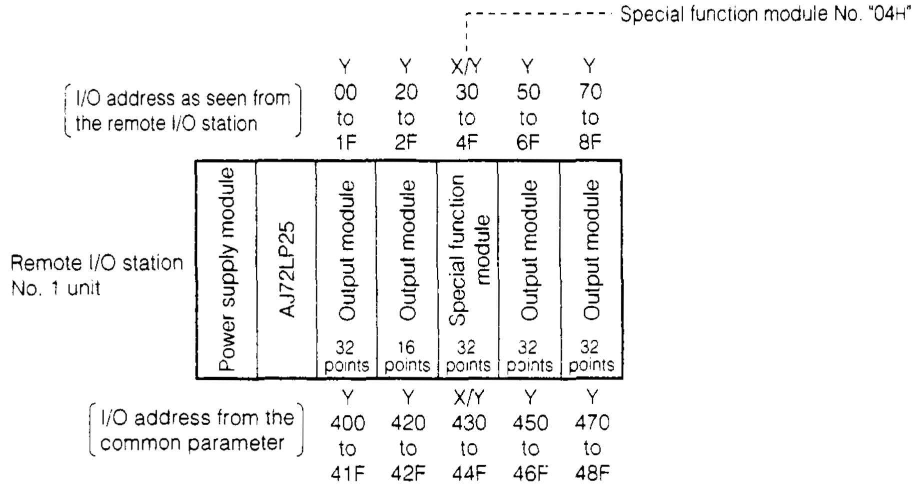
The remote station special function module's module No. is set by the contents of the link parameter set in the master station.

L/R	M←L		M→R	M←R	M→L/R		M←L/R	
No.	B	W	W	W	Y	X/Y	X	Y/X
R1	---	---	29C-309	0F9-15E	400-48F	000-08F	430-44F	030-04F
R2	---	---	215-24F	080-0A3	510-67	010-17F	500-65F	000-15F
R3	---	---	1B6-214	15F-1B5	270-32F	050-10F	220-28F	000-06F
	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-
	-	-	-	-	-	-	-	-



(d) MELSECNET/10 remote I/O station special function module's module No.

For remote I/O station special function module's module Nos., when the final address of all of the following "I/O address as seen from the remote I/O station" are 3 digits then only the first two digits are used. Set using the "I/O addresses as seen from the remote I/O station" regardless of the common parameter contents set in the MELSECNET/10 remote I/O net master station.

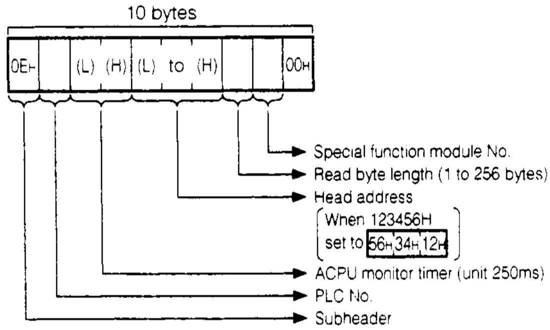


10.4.2 Special Function Module Buffer Memory Read

This section explains the command/response format when reading data from the special function module buffer memory.

1 When exchanging using binary code

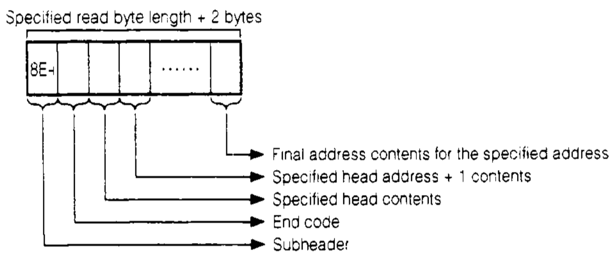
Command format



Remarks

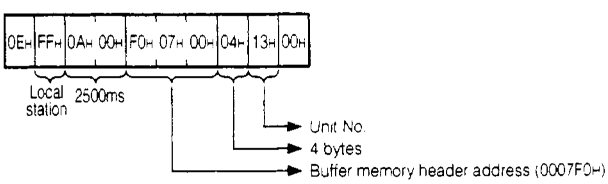
Set to "00H" when specifying the byte length as 256 bytes.

Response format

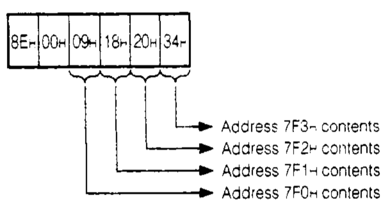


Example: When reading 7F0H to 7F3H of the special function module (X.Y120 to 13F(module No. 13H)) in the same station installed in the E71.

Command (remote node → E71)

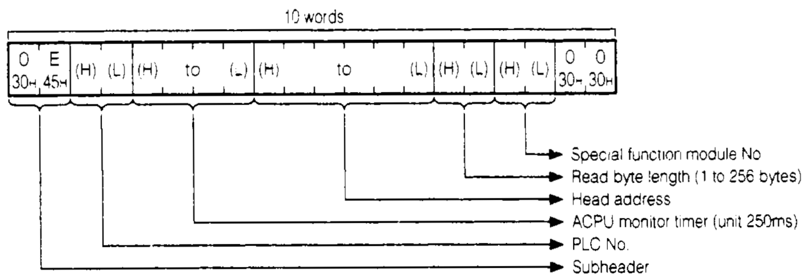


Response (E71 → remote node)



2 When exchanging using ASCII code

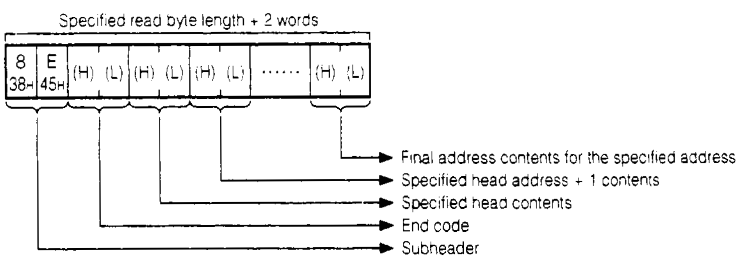
Command format



Remarks

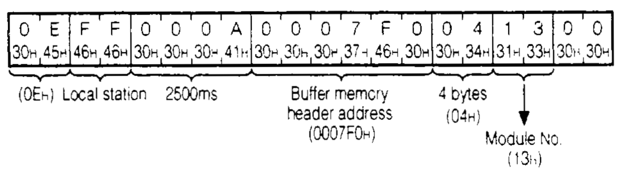
Set to "3030H" when specifying the byte length as 256 bytes.

Response format

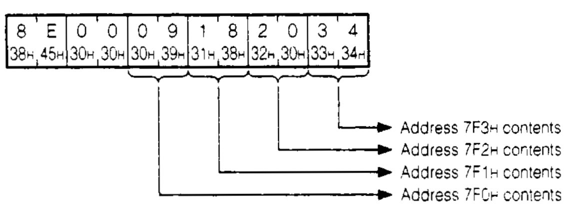


Example: When reading 7F0H to 7F3H of the special function module (X.Y120 to 13F(module No. 13H)) in the same station installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)

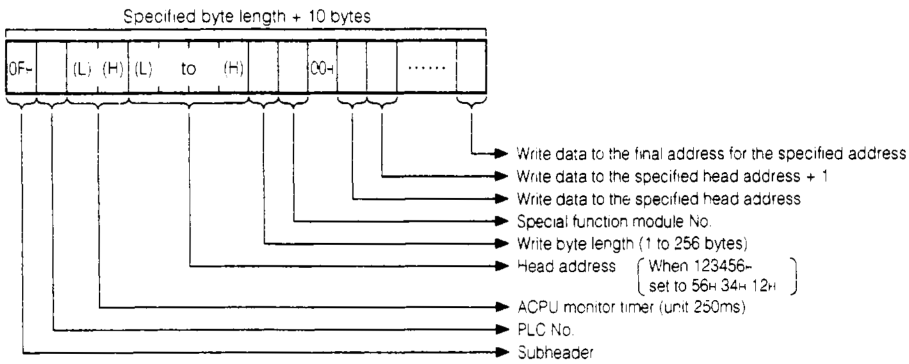


10.4.3 Special Function Module Buffer Memory Write

This section explains the command/response format when writing data to the special function module buffer memory.

1 When exchanging using binary code

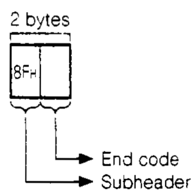
Command format



Remarks

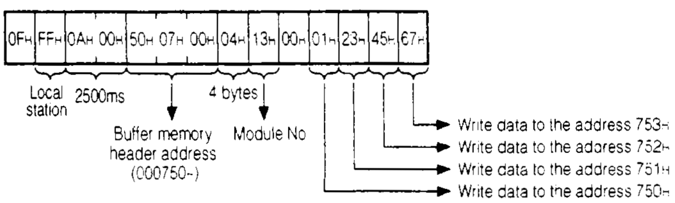
Set to "00_H" when specifying the byte length as 256 bytes.

Response format

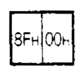


Example: When writing 750_H to 753_H of the special function module (X.Y120 to 13F(module No. 13_H)) in the same station installed in the E71.

Command (remote node → E71)

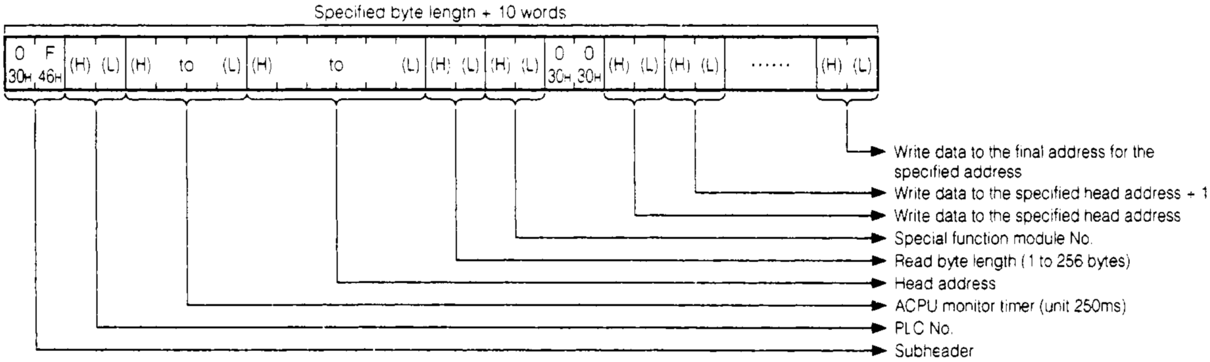


Response (E71 → remote node)



2 When exchanging using ASCII code

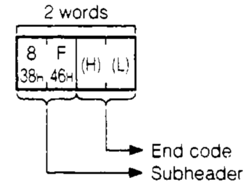
Command format



Remarks

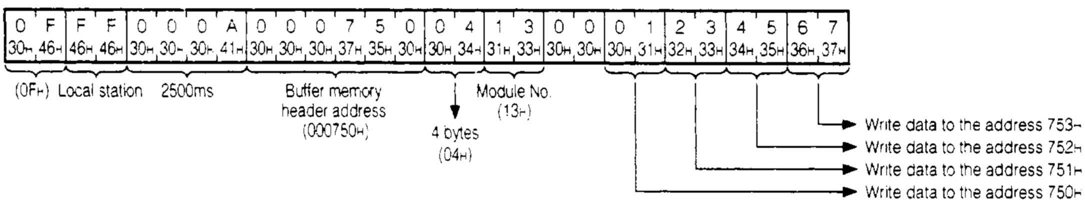
Set to "3030H" when specifying the byte length as 256 bytes.

Response format

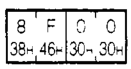


Example: When writing 750H to 753H of the special function module (X.Y120 to 13F(module No. 13H)) in the same station installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)



10.5 Remote RUN/STOP and CPU Model Name Read

This function makes it possible to conduct remote RUN/STOP of the PLC CPU from a remote node and to read what is the model name of the PLC linked to a remote node.

This section explains the control procedure specified contents, method, and example specification when conducting this function.

10.5.1 Commands and Functions

(1) The commands and functions for remote RUN/STOP and PLC model name read are shown in Table 10.5.

Table 10.5. Functions List

Item	Command/ response classification	Processing description	PLC CPU status		
			Stopped	Running: Write possible setting	Running: Write impossible setting
Remote RUN	13_H	Request remote RUN of the PLC CPU.	○	○	○
Remote STOP	14_H	Request remote STOP of the PLC CPU.	○	○	○
PLC model name read	15_H	Reads what the PLC CPU is and whether it is a remote station.	○	○	○

In the PLC CPU status column in the above table the "○" represents execution possible.

Point

(1) When a remote RUN/STOP is conducted for the PLC CPU (local station) installed in the E71, use the data exchange function for when the PLC CPU is stopped and conduct this function. (Refer to Item 5.6)
If the data exchange function for when the PLC CPU is stopped is not used, the initial process request signal (Y19) and the open process request signal (Y8 to F) will turn off when the local station's CPU is stopped, which will make it no longer possible to exchange between the remote node and the E71.

(2) When remote RUN/STOP is conducted for a PLC CPU other than one installed in the E71 (remote station), this function can be executed regardless of whether or not the data communication function for when the PLC CPU is stopped is used.

10.5.2 Remote RUN/STOP

1 Remote RUN/STOP control contents

(a) The PLC CPU status from the remote RUN/STOP from the remote node or the conditions of the RUN/STOP key switch on the front of the PLC CPU are shown in the following table.

		Status of the key switch on the front of the PLC CPU			
		RUN	**STOP**	**PAUSE**	**STEP-RUN**
Specification contents from the remote node	Remote RUN	RUN	STOP	PAUSE	STEP-RUN
	Remote STOP	STOP	STOP	STOP	STOP

Remarks

① The PLC CPU will not enter the RUN status when remote RUN is conducted via local station E71 when the corresponding PLC CPU has already been put in the remote STOP status via a special function module, such as another E71.

② When conducting a remote RUN, whether or not to RUN is determined after the data memory is cleared by the remote relay M9016 and M9017 status.

Special relay		Data memory status
M9016	**M9017**	
OFF	OFF	Run without conducting clear.
OFF	ON	Clear other than the latch range specified by the parameter (However, the link X image is not cleared.)
ON	ON/OFF	RUN after clearing all.

Remarks

When conducting remote RUN as described in the above table and the data memory is not cleared, it is necessary to reset (off) the special relay M9016 and M9017.

Point

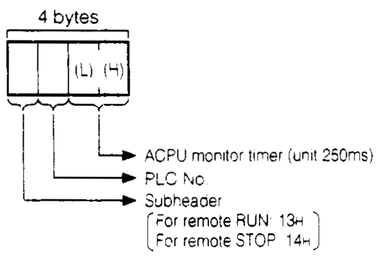
When the power supply has been turned from off to on or the PLC CPU has been reset after a remote RUN/STOP has been conducted from a remote node, delete the remote information.

1 Command/response format

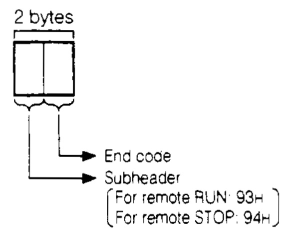
This section explains the command/response format when conducting a PLC CPU remote RUN/STOP from a remote node.

(a) When exchanging using binary code

Command format

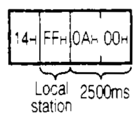


Response format

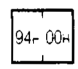


Example: When a remote STOP is conducted for the PLC CPU installed in the E71.

Command (remote node → E71)

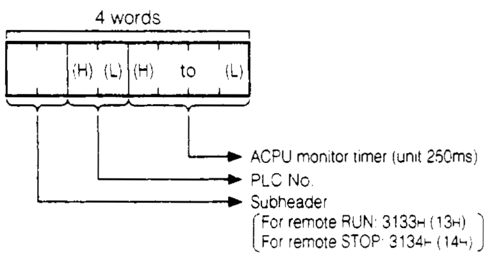


Response (E71 → remote node)

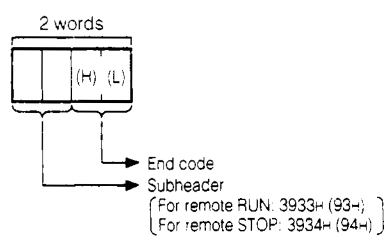


(b) When exchanging using ASCII code

Command format

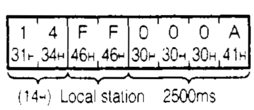


Response format

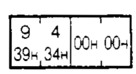


Example: When remote STOP is conducted for a PLC CPU installed in the E71

Command (remote node → E71)



Response (E71 → remote node)



10.5.3 PLC CPU Model Name Read

This function reads the model name of the PLC CPU with which the remote node is communicating via the E71.

1 PLC CPU model name and read code

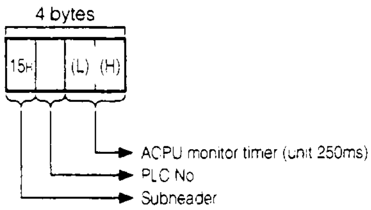
PLC CPU name	Read code (hexadecimal)
A1CPU, A1NCPU	$A1_H$
A2CPU, A2CPU-S1, A2NCPU, A2NCPU-S1, A2SCPU, A2SCPU-S1	$A2_H$
A3CPU, A3NCPU, A1SHCPU, A1SJHCPU, A2SHCPU, A2SHCPU-S1	$A3_H$
A2ACPU, A2UCPU, A2ASCPU, Q2ACPU, Q2ASCPU, Q2ASHCPU	92_H
A2ACPU-S1, A2UCPU-S1, A2ASCPU-S1, Q2ACPU-S1 Q2ASCPU-S1, Q2ASHCPU-S1	93_H
A3ACPU, A3UCPU, A4UCPU, Q3ACPU, Q4ACPU, Q4ARCPU, AJ72LP25/BR15, AJ72QLP25/QBR15	94_H
A0J2HCPU, A1SCPU, A1SCPU-S1, A1SJCPU	98_H
A2CCPU	$9A_H$
AJ72P25/R25	AB_H

2 Command/response format

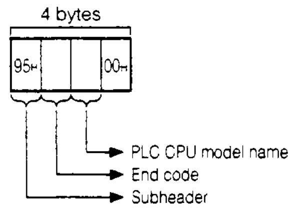
This section explains the command/response format for reading the name of the PLC CPU being used from the remote node.

(a) When exchanging using binary code

Command format

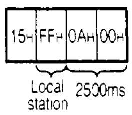


Response format

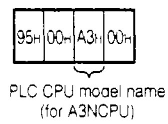


Example: When the model name of a PLC CPU installed in the E71 is read.

Command (remote node → E71)

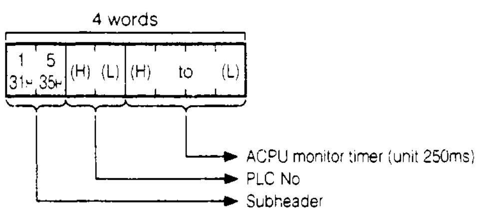


Response (E71 → remote node)

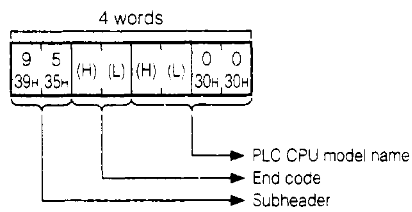


(b) When exchanging using ASCII code

Command format

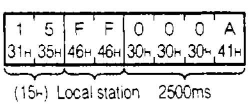


Response format

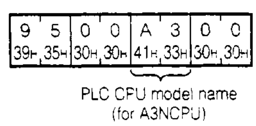


Example: When the model name of a PLC CPU installed in the E71 is read.

Command (remote node → E71)



Response (E71 → remote node)



10.6 Sequence Program Read/Write

This function is used to control the read and storage of various programs (main/sub sequence program, main/sub microcomputer program), parameter data, and comment data to and from the PLC CPU by a remote node; and to write programs, parameter data, and comment data from a remote station in accordance with the control contents to the PLC CPU.

10.6.1 Precautions When Reading/Writing Programs

This section explains precautions when reading/writing programs.

(1) When conducting a program read, read all of the sequence programs, microcomputer programs, parameter data, and comment data areas in the PLC CPU. When writing, write all of the read and stored data to the PLC CPU. If all areas are not written the PLC CPU will not operate correctly.

(2) When writing parameter data, be sure to write it before writing the program and execute an analysis request. If this is not done, the parameters in the PLC CPU user memory will be changed but the parameters stored in the work area used for operation by the PLC CPU will not be changed, so processing will be conducted with the contents before the change (contents stored in the work area) even when the peripheral equipment is installed and operated after the change.

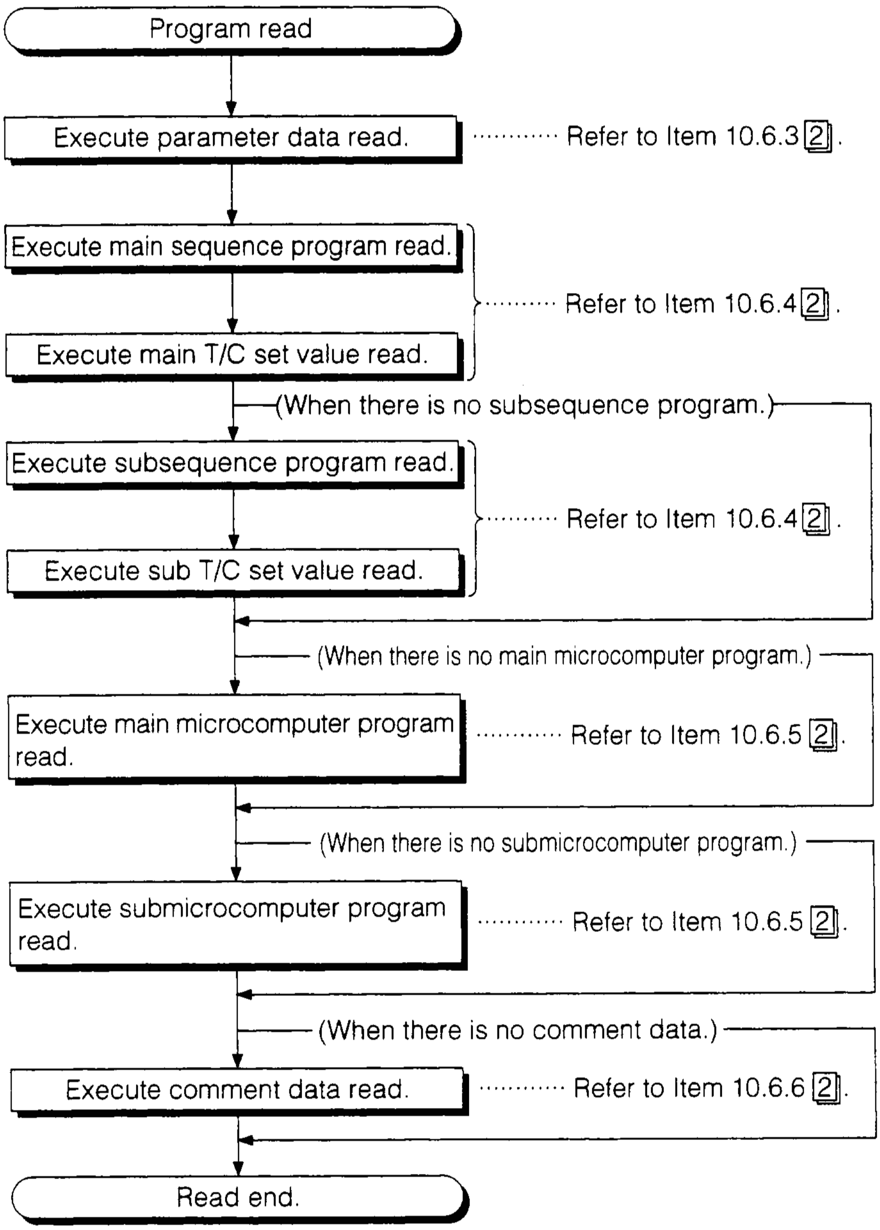
(3) The commands determine the number of processes that can be executed for one exchange. When conducting read or write, divide the data into several pieces and read or write all areas.

(4) Conduct read/write for the A4UCPU subsequence program for the sub 1. Read and write cannot be conducted for sub 2 to sub 4.

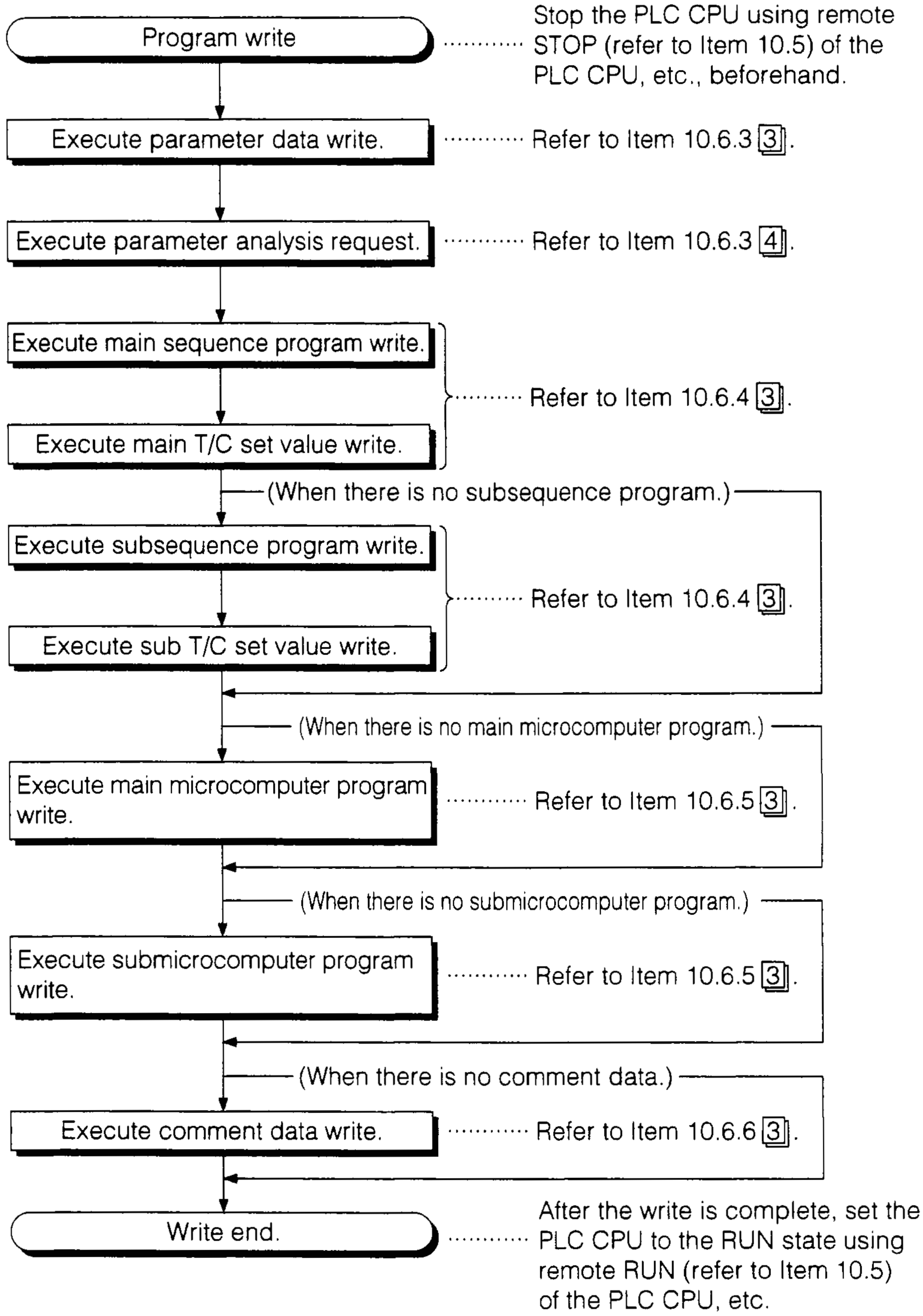
10.6.2 Program Read/Write

This section explains the processing procedure when conducting program read/write.

1 Read procedure



2 Write procedures



10.6.3 Parameter Memory Read, Write, and Analysis Request

This section explains the control procedure specification contents, method, and example specification when reading or writing the parameter memory contents of the PLC CPU.

1 Commands and addresses

(a) The functions used to read/write parameters are shown in Table 10.6

Table 10.6 Functions List

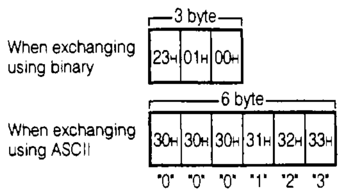
Item	Command/ response classification	Processing description	Number of processing points conducted in one exchange	PLC CPU status		
				Stopped	Running	
					Write possible setting	Write impossible setting
Batch read	10_H	Reads the PLC CPU parameter contents.	256 bytes	○	○	○
Batch write	11_H	Writes the PLC CPU parameter contents.		○	×	×
Analysis request	12_H	Causes the PLC CPU to recognize and check the switching parameter contents.		○	×	×

In the PLC CPU status column in the above table the "○" represents executable and the "×" represents not executable.

(b) Parameter addresses

The parameter memory area is the 3k bytes from 0_H to BFF_H. As shown below the address specification is 3 bytes when exchanging using binary and 6 bytes when exchanging using ASCII.

Example: When specifying address 123_H.



Point

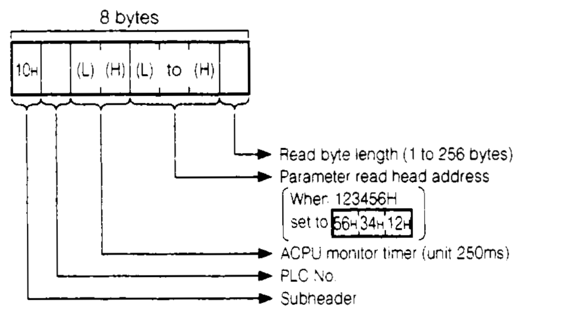
When changing the parameter memory contents, be sure to conduct a parameter analysis request after writing all of the data to be changed.
If this is not done the parameters in the CPU user memory will be changed but the parameter contents stored in the work area used by the PLC CPU for operation will not be changed, so processing will be conducted using the parameter contents before the change (contents stored in the work area) even if the peripheral equipment is installed and operated after the change.

2 Batch read

This section explains the command/response format when reading the PLC CPU parameter memory contents.

(a) When exchanging using binary code

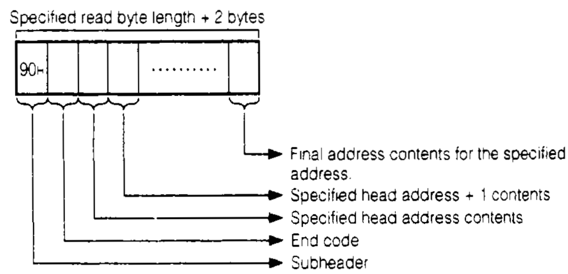
Command format



Remarks

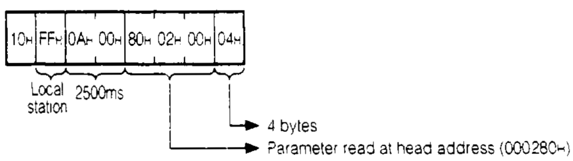
Set to "00H" when specifying the byte length as 256 bytes.

Response format

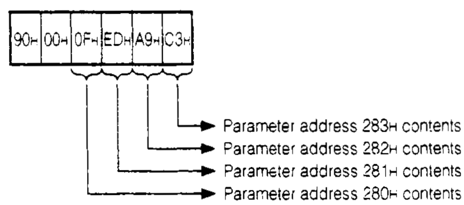


Example: When reading the parameter data (280H to 283H) of the PLC CPU installed in the E71

Command (remote node → E71)

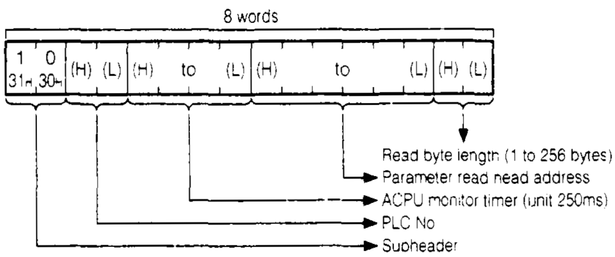


Response (E71 → remote node)



(b) When exchanging using ASCII code

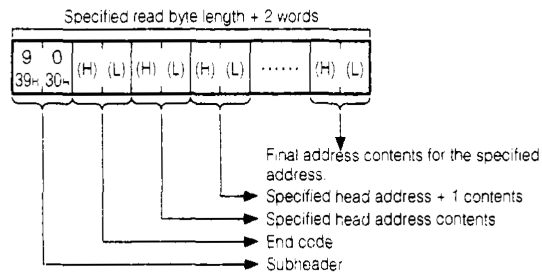
Command format



Remarks

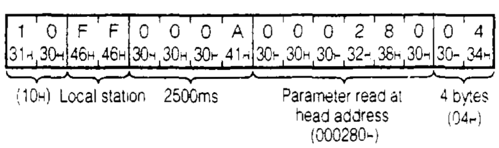
Set to "3030H" when specifying the byte length as 256 bytes.

Response format

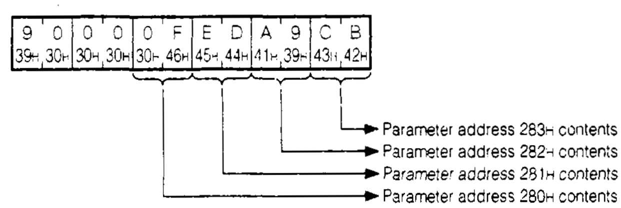


Example: When reading the parameter data (280H to 283H) of the PLC CPU installed in the E71

Command (remote node → E71)



Response (E71 → remote node)

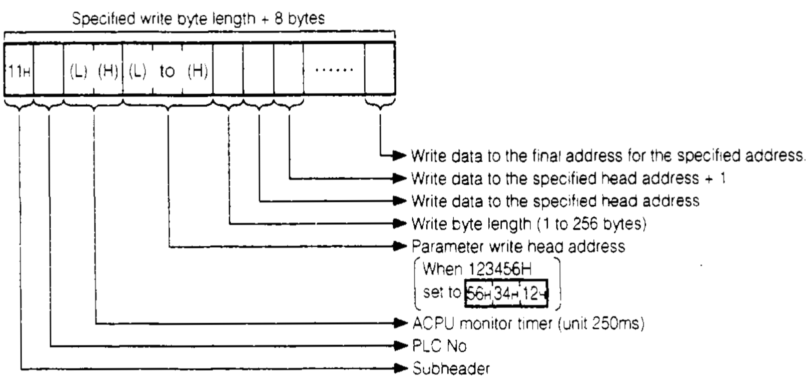


3 Batch write

This section explains the command/response format when writing data to the PLC CPU buffer memory.

(a) When exchanging using binary code

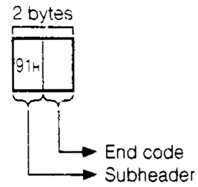
Command format



Remarks

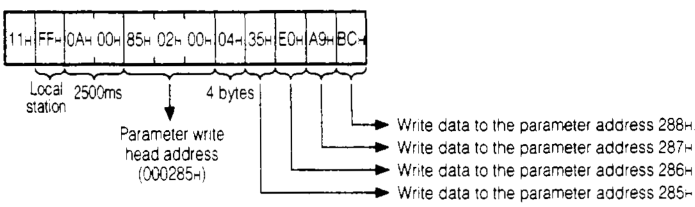
Set to "00H" when specifying the byte length as 256 bytes.

Response format

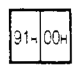


Example: When writing data to the parameter memory (285H to 288H) of the PLC CPU installed in the E71.

Command (remote node → E71)

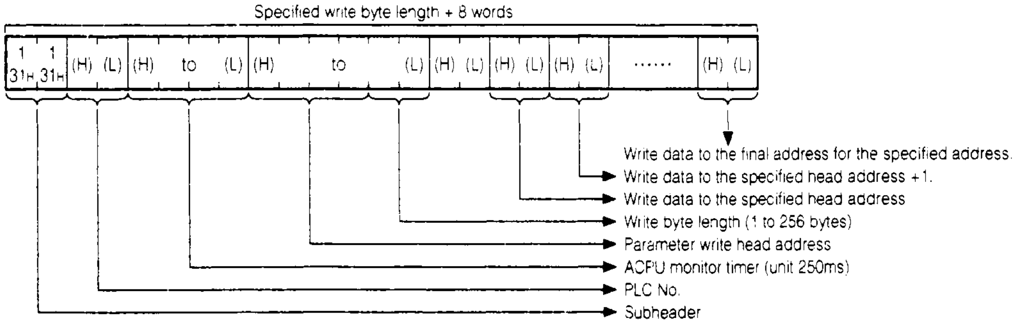


Response (E71 → remote node)



(b) When exchanging using ASCII code

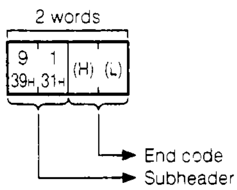
Command format



Remarks

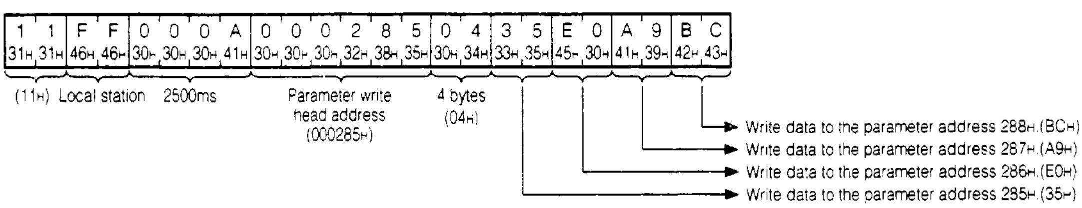
Set to "3030H" when specifying the byte length as 256 bytes.

Response format

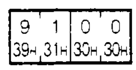


Example: When writing data to the parameter memory (285H to 288H) of the PLC CPU installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)

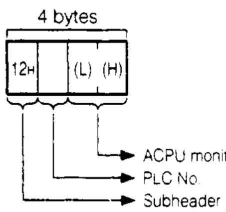


4 Analysis request

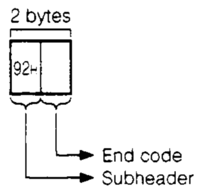
This section explains the command/response format when conducting a parameter data analysis request of the PLC CPU. The analysis request makes the PLC CPU recognize the parameter changes conducted when the parameter data is changed and is a command that causes the changed parameters to be written to the CPU. If analysis request is not conducted the PLC CPU will not operate with the changed parameters.

(a) When exchanging using binary code

Command format

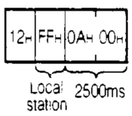


Response format

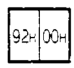


Example: When conducting parameter analysis request of the PLC CPU installed in the E71.

Command (remote node → E71)

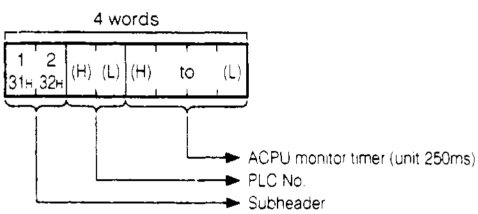


Response (E71 → remote node)

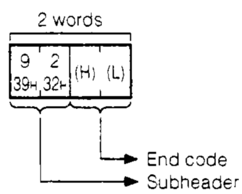


(b) When exchanging using ASCII code

Command format

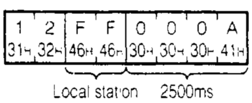


Response format

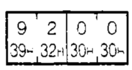


Example: When conducting a parameter analysis request of the PLC CPU installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)



10.6.4 Sequence Program Read/Write

This section explains the control procedure specification contents, method, and example specification when reading and writing the PLC CPU's sequence program.

1 Commands and setting method

(a) The functions used for reading/writing the sequence program are shown in Table 10.7.

Table 10.7 Function List

Item			Command/ response classification	Processing description	Number of processing points conducted in one exchange	PLC CPU status		
						Stopped	Running: Write possible setting	Running: Write impossible setting
Batch read	Main	Program	$0A_H$	Reads the main sequence program.	256 steps	○	○	○
		T/C set value		Reads the T/C set values used by the main sequence program.	256 points			
	Sub	Program	$0B_H$	Reads the sub sequence program.	256 steps	○	○	×
		T/C set value		Reads the T/C set values used by the subsequence program.	256 points			
Batch write	Main	Program	$0C_H$	Writes the main sequence program.	256 steps	○	○*	×
		T/C set value		Writes the T/C set values used by the main sequence program.	256 points	○	○	×
	Sub	Program	$0D_H$	Writes the subsequence program.	256 steps	○	○*	×
		T/C set value		Writes the T/C set values used by the subsequence program.	256 points	○	○	×

In the PLC CPU status column in the above Table the "○" represents execution possible and the "×" represents execution not possible.

* All of the following conditions must be met when conducting program write during RUN.

① The PLC CPU is an A3, A3N, A3A, A3U, or A4U

② A program that is not operating. (Shows subprogram when the main program is running.)

③ The PLC CPU special relay is in the following state.

ⓐ M9050 (Signal flow replacement point) .... Off (A3CPU only)

ⓑ M9051 (CHG instruction execution prohibited) .. On

* The A4U subsequence program read/write is conducted for sub 1. Read/write is not conducted for sub 2 to sub 4.

(b) Sequence program step No. specification

The sequence program step Nos. are specified using hexadecimal numbers as shown in Table 10.8.

Table 10.8 Step Nos.

Step No.	Set value
Step 0	0000_H
Step 1	0001_H
to	to
Step 30719 (30k)	$77FE_H$

(c) Device No. specification when reading/writing T/C set values

The device Nos. used when reading/writing T/C set values are set using the codes shown in Table 10.9. Reading/writing of T/C set values is done in the range of T0 to 255 and C0 to 255. The T256 to 2047 and C256 to 1023 set values cannot be read/written. To read/ write setting values, conduct a device memory read/write.

Table 10.9 T/C Set Value Specification

Device No.	Setting code
T0's set value T1's set value to T255's set value	$FE00_H$ $FE01_H$ to $FEFF_H$
C0's set value C1's set value to C255's set value	$FF00_H$ $FF01_H$ to $FFFF_H$

The relationship between the device No. and the setting code is shown below.

Timer: Tm = FE00H + n
Counter: Cm = FF00H + n
m : Device No.
n : The device No. is converted into a hexadecimal value.

(d) T/C set value contents

The T/C set value is exchanged in hexadecimal numbers as shown in Table 10.10. When writing over the T/C set values from a remote node via a E71, specify the setting data shown in Table 10.10.

Example: Setting data when the T10's set value K10 is changed to K20 .... 0014H
Setting data when the T11's set value D30 is changed to D10 .... 8014H

Table 10.10 T/C Set Value Data Specification

Example circuit in the program	Setting contents in the program	Set value
K□□□□□ —(T□□□)— K□□□□□ —(C□□□)—	K0 K1 to K9 K10 to K32767	0000_H 0001_H to 0009_H $000A_H$ to $7FFF_H$
D□□□□ —(T□□□)— D□□□□ —(C□□□)—	D0 D1 D2 to D1023	8000_H 8002_H 8004_H to $87FE_H$

The relationship between the setting contents and setting data in the program is as follows.

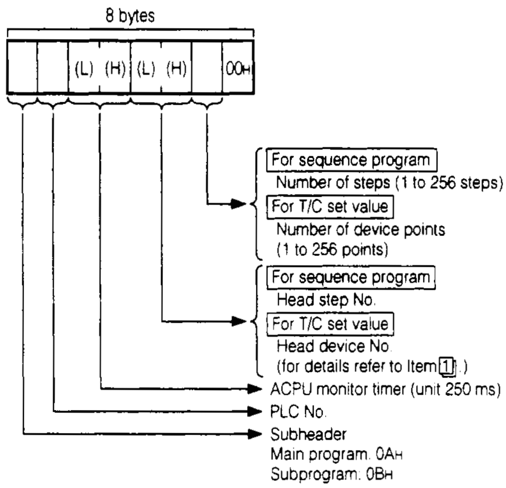
$Km = 0000_H + n$
$Dm = 8000_H + 2n$
m : Device No.
n : The device No. is converted into a hexadecimal number value.

2 Batch read

This section explains the command/response format when batch reading the sequence program contents (machine language), timer (T), and counter (C) set values.

(a) When exchanging using binary code

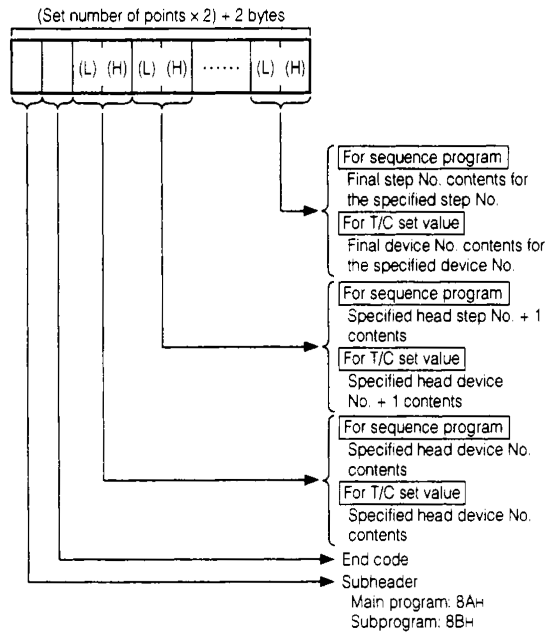
Command format



Remarks

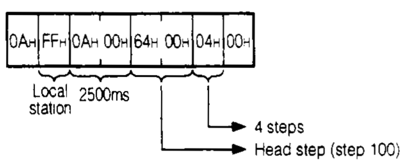
Set to "00H" when setting the number of steps and number of device points to 256 points.

Response format

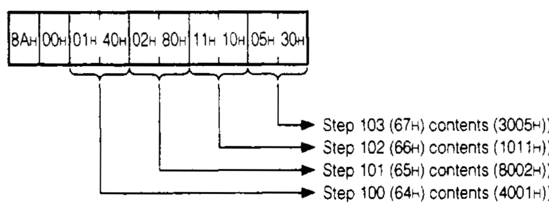


Example 1: When reading the main sequence program (100 to 103 steps) for the PLC CPU installed in the E71.

Command (remote node → E71)

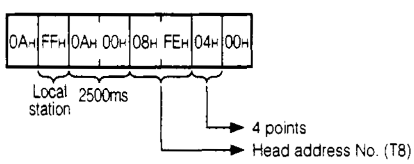


Response (E71 → remote node)

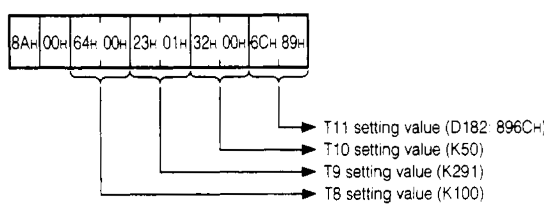


Example 2: When directly reading the timer set value (T8 to T11) used by the main sequence program of the PLC CPU installed in the E71.

Command (remote node → E71)

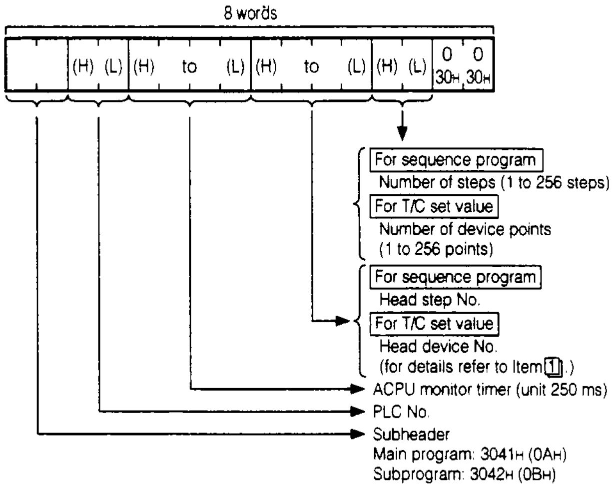


Response (E71 → remote node)



(b) When exchanging using ASCII code

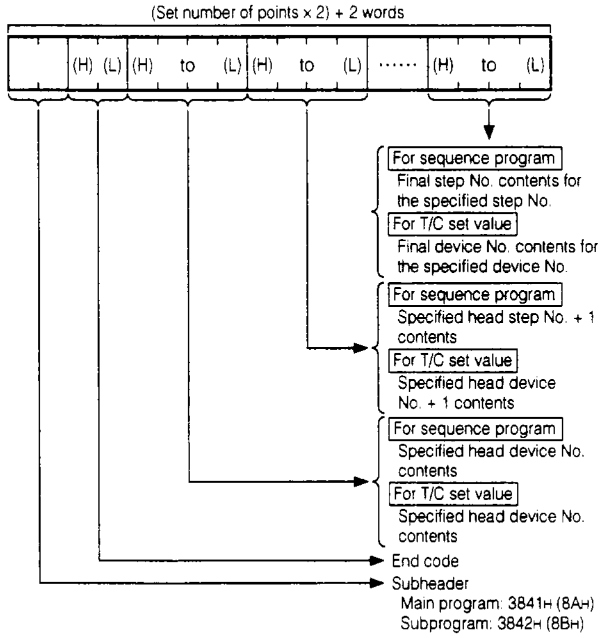
Command format



Remarks

Set to "3030H" when setting the number of steps and number of device points to 256 points.

Response format



Example 1: When reading the main sequence program (100 to 103 steps) for the PLC CPU installed in the E71.

Command (remote node → E71)

0 A	F F	0 0 0 A	0 0 6 4	0 4	0 0
30H, 41H	46H, 46H	30H, 30H, 30H, 41H	30H, 30H, 36H, 34H	30H, 34H	30H, 30H
(0AH)	Local station	2500ms	Head step No. (step 100) (0064H)	4 steps (04H)	

Response (E71 → remote node)

8 A	0 0	4 0 0 1	8 0 0 2	1 0 1 1	3 0 0 5
38H, 41H	30H, 30H	34H, 30H, 30H, 31H	38H, 30H, 30H, 32H	31H, 30H, 31H, 31H	33H, 30H, 30H, 35H
		Step 100 (64H) contents (4001H)	Step 101 (65H) contents (8002H)	Step 102 (66H) contents (1011H)	Step 103 (67H) contents (3005H)

Example 2: When directly reading the timer set value (T8 to T11) used by the main sequence program of the PLC CPU installed in the E71.

Command (remote node → E71)

0 A	F F	0 0 0 A	F E 0 8	0 4	0 0
30H, 41H	46H, 46H	30H, 30H, 30H, 41H	46H, 45H, 30H, 38H	30H, 34H	30H, 30H
(0AH)	Local station	2500ms	Head device No. (T8) (FE08H)	4 points (04H)	

Response (E71 → remote node)

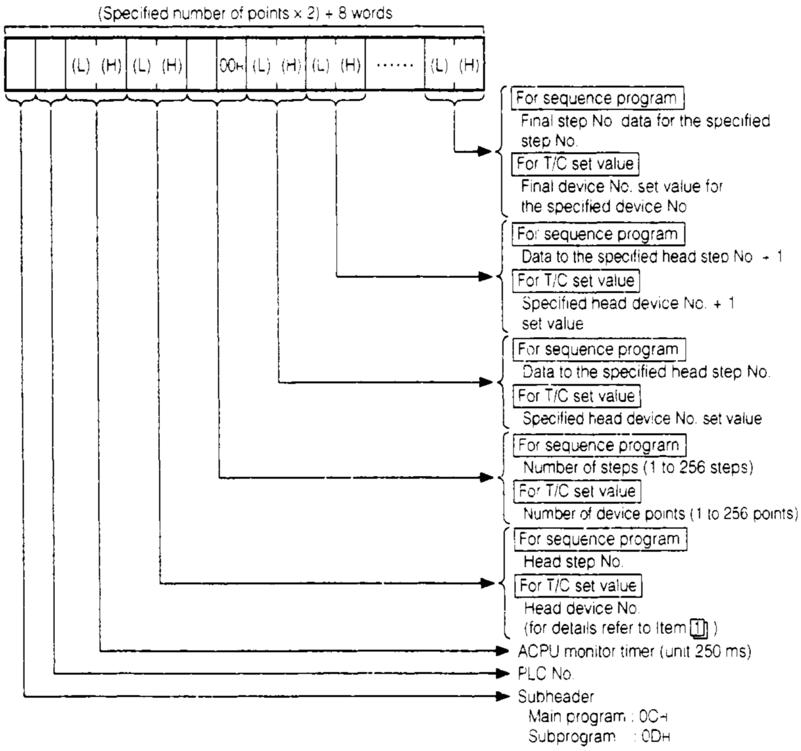
8 A	0 0	0 0 6 4	0 1 2 3	0 0 3 2	8 9 6 C
38H, 41H	30H, 30H	30H, 30H, 36H, 34H	30H, 31H, 32H, 33H	30H, 30H, 33H, 32H	38H, 39H, 36H, 43H
		T8 set value (0064H: K100)	T9 set value (0123H: K291)	T10 set value (0032H: K50)	T11 set value (896CH: D182)

3 Batch write

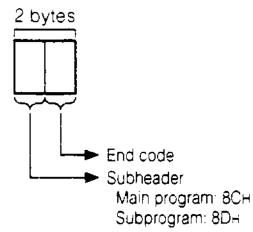
This section explains the command/response format when batch writing the set values for the sequence program contents (machine language), timer (T), and counter (C).

(a) When exchanging using binary code

Command format



Response format

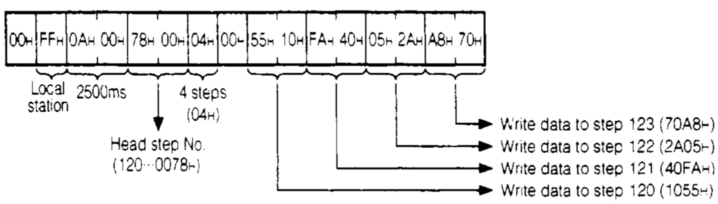


Remarks

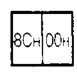
Set "00H" when specifying the number of steps and number of device points as 256 points.

Example 1: When writing the main sequence program (steps 120 to 123) for the PLC CPU installed in the E71.

Command (remote node → E71)

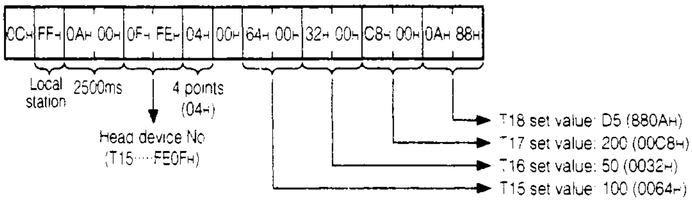


Response (E71 → remote node)

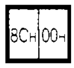


Example 2: When changing the set value of the timer (T15 to T18) used by the main sequence program of the PLC CPU installed in the E71.

Command (remote node → E71)

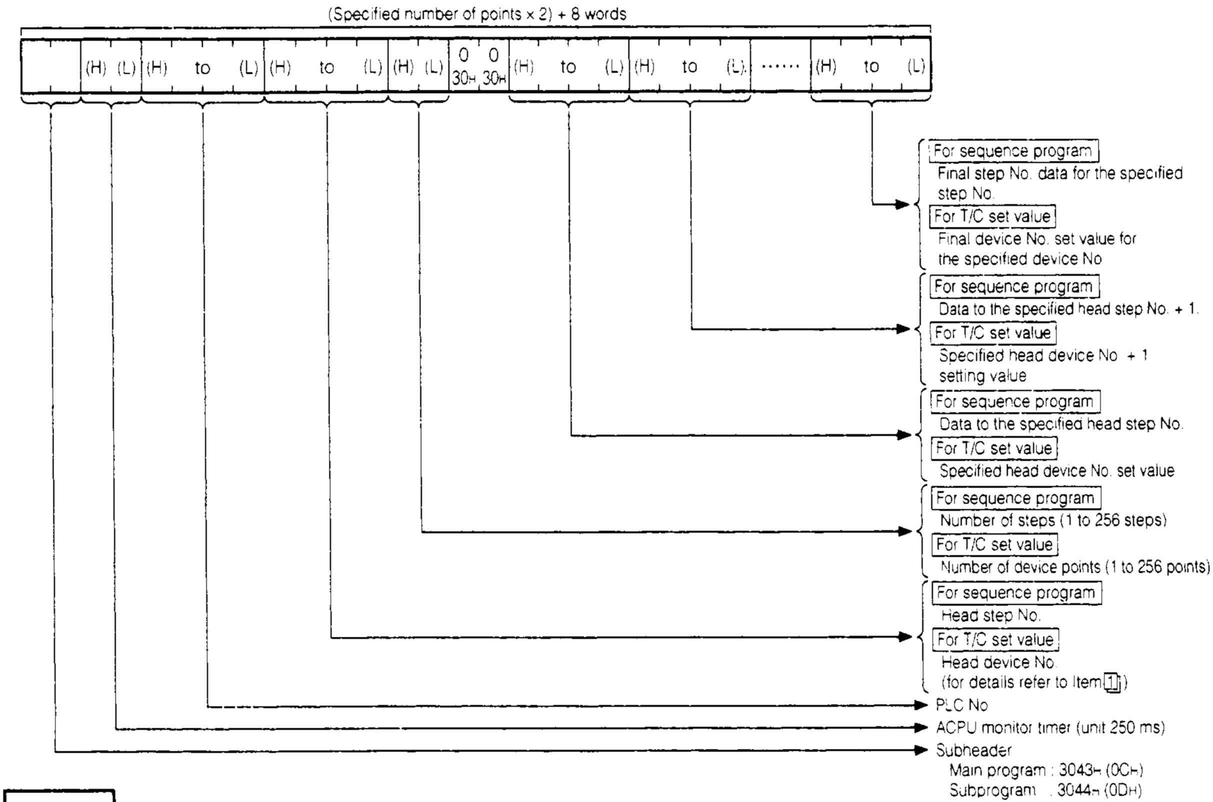


Response (E71 → remote node)



(b) When exchanging using ASCII code

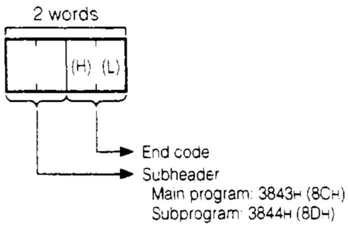
Command format



Remarks

Set to "3030H" when specifying the number of steps and number of devices as 256 points.

Response format



Example 1: When writing the main sequence program (steps 120 to 123) for the PLC CPU installed in the E71.

Command (remote node → E71)

0 C	F F	0 0 0 A	0 0 7 8	0 4	0 0	1 0 5 5	4 0 F A	2 A 0 5	7 0 A 8
30H 43H	46H 46H	30H 30H 30H 41H	30H 30H 37H 38H	30H 34H	30H 30H	31H 30H 35H 35H	34H 30H 46H 41H	32H 41H 30H 35H	37H 30H 41H 38H
(0CH)	Local station	2500ms	Head step No (120·····0078H)	4 steps (04H)		Write data to step 120 (1055H)	Write data to step 121 (40FAH)	Write data to step 122 (2A05H)	Write data to step 123 (70A8H)

Response (E71 → remote node)

8 C	0 0
38H 43H	30H 30H

Example 2: When changing the set value of the timer (T15 to T18) used by the main sequence program of the PLC CPU installed in the E71.

Command (remote node → E71)

0 C	F F	0 0 0 A	F E 0 F	0 4	0 0	0 0 6 4	0 0 3 2	0 0 C 8	8 8 0 A
30H 43H	46H 46H	30H 30H 30H 41H	46H 45H 30H 46H	30H 34H	30H 30H	30H 30H 36H 34H	30H 30H 33H 32H	30H 30H 43H 38H	38H 38H 30H 41H
(0CH)	Local station	2500ms	Head device No (T15·····FE0FH)	4 points (04H)		T15 set value: 100 (0064H)	T16 set value: 50 (0032H)	T17 set value: 200 (00C8H)	T18 set value: D5 (880AH)

Response (E71 → remote node)

8 C	0 0
38H 43H	30H 30H

10.6.5 Microcomputer Program Read/Write

This section explains the control procedure specification contents, method, and example specification when reading/writing a PLC CPU microcomputer program.

1 Commands and addresses

This section explains the command/response classification and program addresses when reading/writing microcomputer programs.

(a) The functions used to read/write microcomputer programs are shown in Table 10.11.

Table 10.11 Functions List

Item		Command/ response classification	Processing description	Number of processing points conducted in one exchange	PLC CPU status		
					Stopped	Running: Write possible setting	Running: Write impossible setting
Batch read	Main	$1E_H$	Reads the main sequence microcomputer program.	256 bytes	○	○	○
	Sub	$1F_H$	Reads the subsequence microcomputer program.				
Batch write	Main	20_H	Writes the main sequence microcomputer program.		○	○*	×
	Sub	21_H	Writes the subsequence microcomputer program.				

In the PLC CPU status column in the above table the "○" represents execution possible and the "×" represents execution not possible.

- All of the following conditions must be met in order to conduct program write during RUN.
  ① PLC CPU is an A3, A3N, A3A, A3U, or A4U.
  ② A program that is not operating. (Shows the sub program if the main program is running.)
  ③ The PLC CPU special relay is in the following status.
    ⓐ M9050 (Signal flow replacement point) .... Off (A3CPU only)
    ⓑ M9051 (CHG instruction execution prohibited) .... On

Point

When the PLC CPU is an AnA/AnUCPU, the SFC program reads/writes the main microcomputer program using the read/write functions. (Write cannot be done while the PLC CPU is running.) When reading from or writing to the SFC program, conduct the read/write in the microcomputer program capacity and microcomputer program address range shown in 1 (b). The microcomputer program capacity is the capacity that is set in the GPP function memory capacity setting or the MELSAP-II function's SFC area capacity setting.

(b) Microcomputer program addresses

The microcomputer program addresses that are specified by the control procedures are conducted by the contents shown below.

① The address range that can be specified by each CPU is shown in the following table.

CPU name	Microcomputer program capacity	Microcomputer program addresses
A1SCPU (S1) A1SJCPU A1SHCPU A1SJHCPU A0J2HCPU A2CCPU A2CJCPU	Maximum 14k bytes	0000_H to $37FE_H$
A1CPU A1NCPU	Maximum 10k bytes	0000_H to $27FE_H$
A2SCPU (S1) A2SHCPU (S1) A2ASCPU (S1) A2CPU (S1) A2NCPU (S1) A2ACPU (S1) A2UCPU (S1)	Maximum 26k bytes	0000_H to $67FE_H$
A3CPU A3NCPU A3ACPU A3UCPU A4UCPU	Maximum 58k bytes for both main and sub	0000_H to $E7FE_H$

② When exchanging using ASCII code, the address is converted to ASCII code with a four digit hexadecimal number.

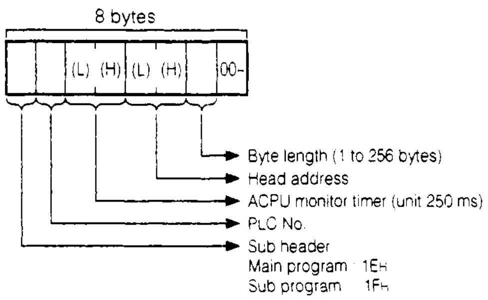
③ When the head address + number of bytes – 1 <= is not the microcomputer program capacity, an error (End error 57_H) will occur.

2 Batch read

This section explains the command/response format when batch reading the microcomputer program contents.

(a) When exchanging using binary code

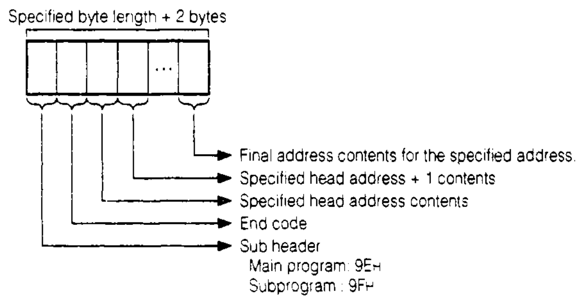
Command format



Remarks

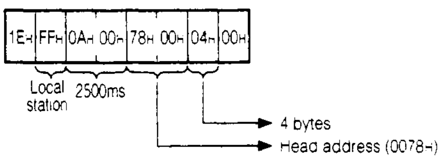
Set to "00H" when specifying the byte length as 256 bytes.

Response format

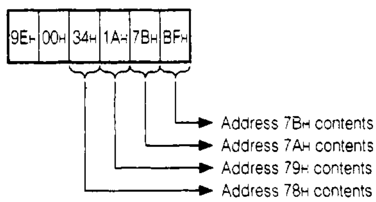


Example: When reading the microcomputer program (78H to 7BH) of the PLC CPU installed in the E71.

Command (remote node → E71)

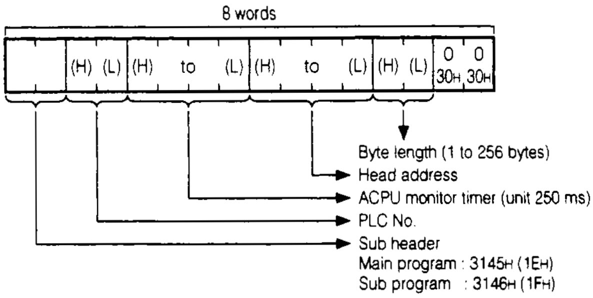


Response (E71 → remote node)

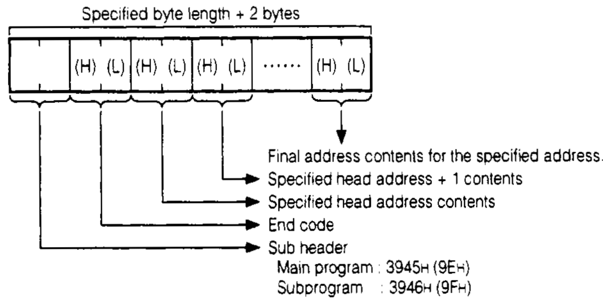


(b) When exchanging using ASCII code

Command format



Response format

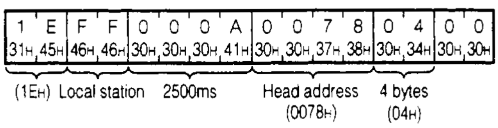


Remarks

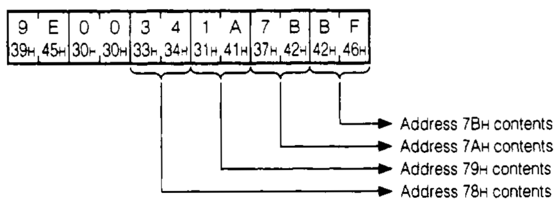
Set to "3030H" when specifying the byte length as 256 bytes.

Example: When reading the microcomputer program (78H to 7BH) of the PLC CPU installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)

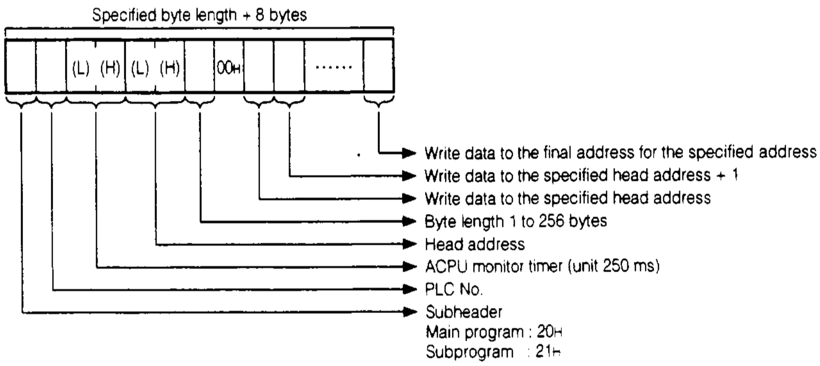


3 Batch write

This section explains the command/response format when batch writing the contents of the microcomputer program.

(a) When exchaaging using binary code

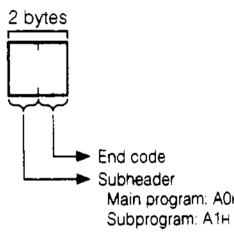
Command format



Remarks

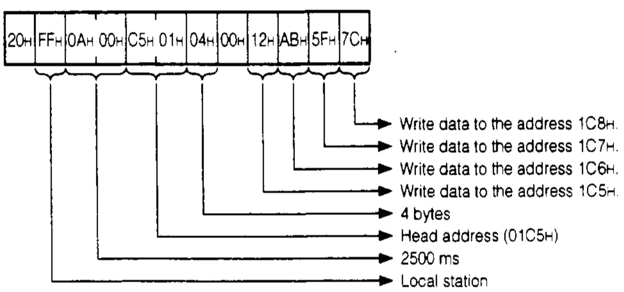
Set to "00H" when specifying the byte length to 256 bytes.

Response format

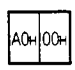


Example: When writing the main microcomputer program (1C5H to 1C8H) of the PLC CPU installed in the E71

Command (remote node → E71)

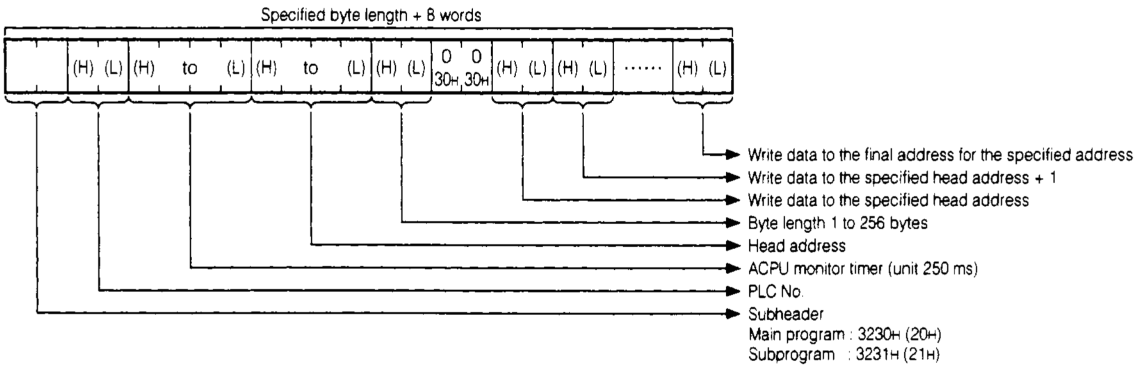


Response (E71 → remote node)



(b) When exchanging using ASCII code

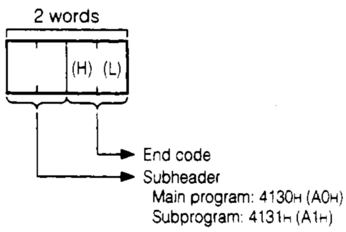
Command format



Remarks

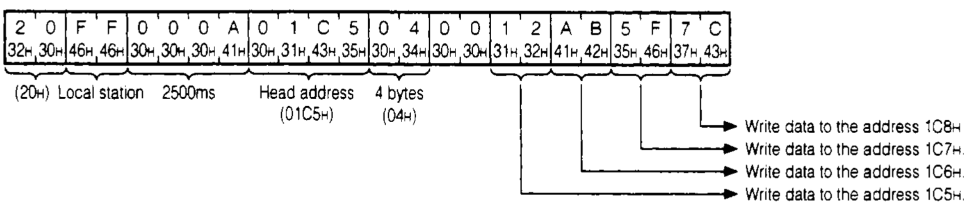
Specify as "3030H" when specifying the byte length as 256 bytes.

Response format

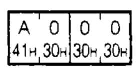


Example: When writing the main microcomputer program (1C5H to 1C8H) to the PLC CPU installed in the P71.

Command (remote node → E71)



Response (E71 → remote node)



10.6.6 Comment Read/Write

This section explains the control procedure specification contents, method, and example specification when reading/writing PLC CPU comment data.

1 Commands and addresses

This section explains the command/response classification and comment data addresses when reading/writing comment data.

(a) The functions used to read/write comment data are shown in Table 10.12.

Table 10.12 Function List

Item	Command/ response classification	Processing description	Number of processing points con-ducted in one exchange	PLC CPU status		
					Running	
				Stopped	Write possible setting	Write impossible setting
Batch read	$1C_H$	Reads the comment memory contents.	256 bytes	○	○	○
Batch write	$1D_H$	Writes the data in the comment memory.		○	○	×

In the status column for the PLC CPU in the above table the capital "○" represents execution possible and the capital "×" represents execution not possible.

(b) Comment memory address

The comment data storage area can be controlled using the corresponding address for a head address of 00_H. For example, if the parameter comment capacity is 2k bytes, the range that can be specified by the head address is 00_H to $7FF_H$.

① The comment memory has a maximum capacity of 64k bytes. The comment data address range is determined by the parameter setting capacity.

② The comment memory address specification is done in hexadecimal numbers.

③ If the head address + specified number of bytes – 1 <= is not the comment memory capacity, an error (end code 57_H) will occur.

Point

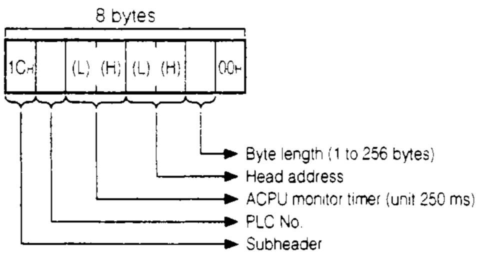
The comment data cannot be read or written by specifying the special device or device No. Be sure to read/write all the data from 0_H.

2 Batch read

This section explains the command/response format when conducting comment memory batch read.

(a) When exchanging with binary code

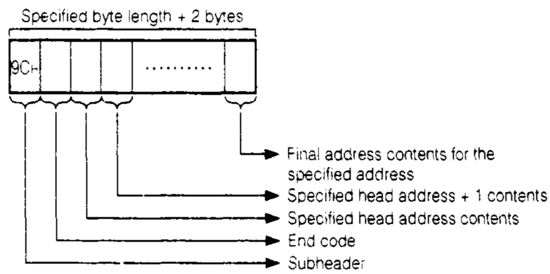
Command format



Remarks

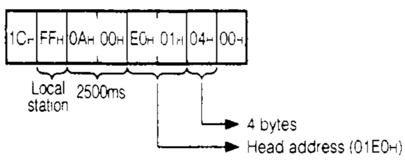
Set to "00H" when specifying the byte length as 256 bytes.

Response format

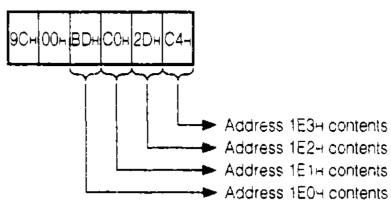


Example: When reading the comment (1E0H to 1E3H) for the PLC CPU installed in the E71.

Command (remote node → E71)

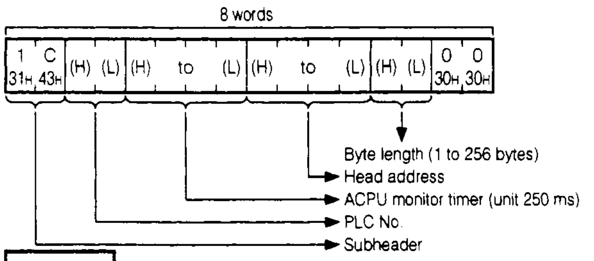


Response (E71 → remote node)



(b) When exchanging using ASCII code

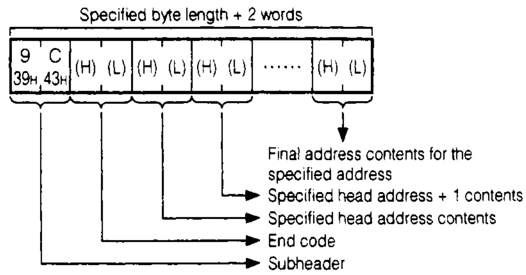
Command format



Remarks

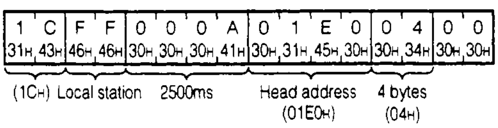
Set to "3030H" when specifying the byte length as 256 bytes.

Response format

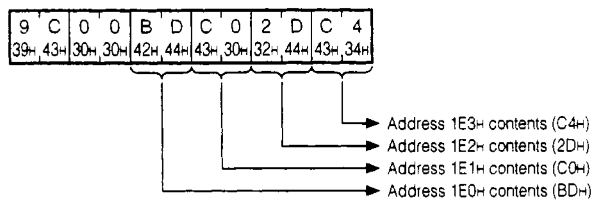


Example: When reading the comment ($1E0_H$ to $1E3_H$) of the PLC CPU installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)

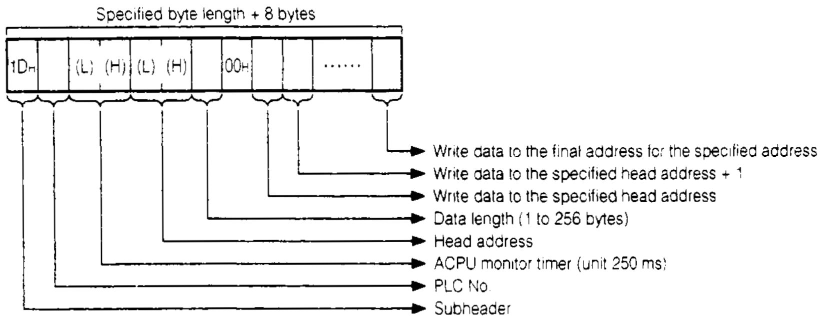


3 Batch write

This section explains the command/response format when batch writing comment memory.

(a) When exchanging using binary code

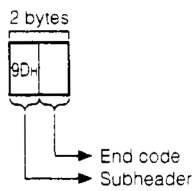
Command format



Remarks

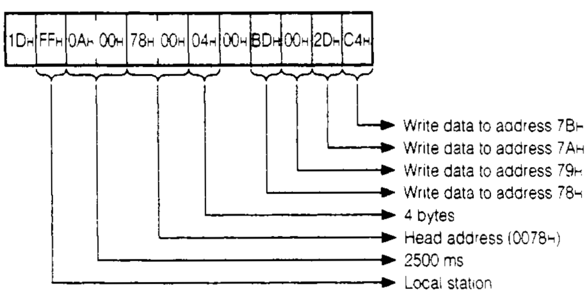
Specify to "00H" when specifying the byte length as 256 bytes.

Response format

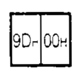


Example: When writing the comment area (78H to 7BH) data to the PLC CPU installed in the E71.

Command (remote node → E71)

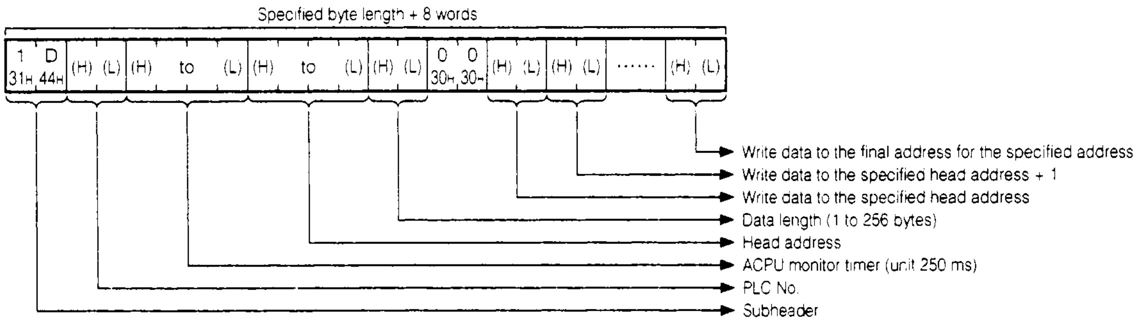


Response (E71 → remote node)



(b) When exchanging using ASCII code

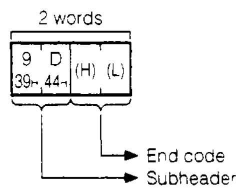
Command format



Remarks

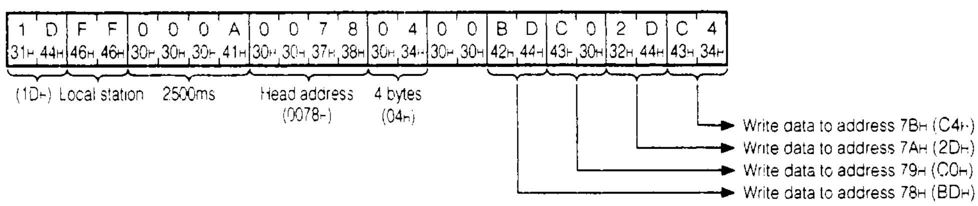
Set to "3030H" when specifying the byte length as 256 bytes.

Response format

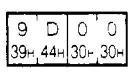


Example: When writing the comment area (78H to 7BH) data of the PLC CPU installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)



10.6.7 Extension Comment Read/Write

This section explains the control procedure specification contents, method contents, and example specification when reading/writing PLC CPU extension comment data.

1 Command and addresses

This section explains the command/response classification and extension comment data addresses when reading/writing extension comment data.

(a) For functions used to read/write extension comment data are shown in Table 10.13.

Table 10.13 Functions List

Item	Command/ response classification	Processing description	Number of processing points conducted in one exchange	PLC CPU status		
				Stopped	Running: Write possible setting	Running: Write impossible setting
Batch read	39_H	Reads the extension comment memory contents.	256 bytes	○	○	○
Batch write	$3A_H$	Writes the data to the extension comment memory.		○	○	×

In the PLC CPU status column in the above table the capital "○" represents execution possible and the capital "x" represents execution not possible.

(b) Extension comment memory address

Extension comment data storage area can be managed by using the corresponding address for the head address 00_H. For example, when the parameter extension comment capacity is 2k bytes, the range that can be sent by the head address is 00_H to $7FF_H$.

① The maximum capacity of the extension comment memory is 63k bytes. The extension comment data address range is determined by the parameter setting capacity.

② Extension comment memory address specification is done using hexadecimal numbers.

③ If the head address + specified number of bytes – 1 <= is not the comment memory capacity, an error (end code 57_H) will occur.

Point

(1) Read/write of the extension comment data cannot be done by specifying the special device or device No. Be sure to read all of the data from 0_H when reading/writing.

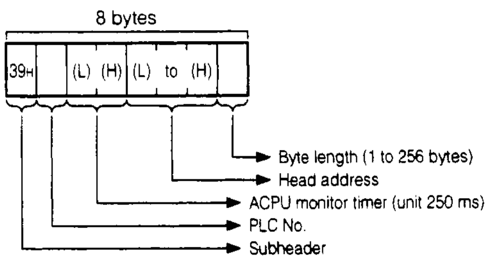
(2) Reading/writing of extension comments can only be conducted for the AnACPU and the AnUCPU.

2 Batch read

This section explains the command/response format for batch reading form the extension comment memory.

(a) When exchanging using binary code

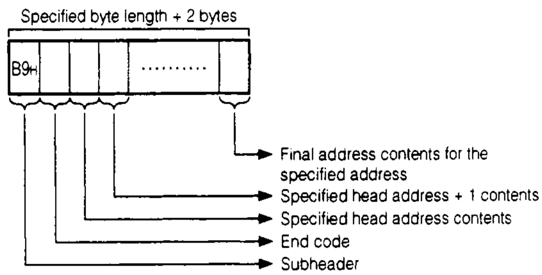
Command format



Remarks

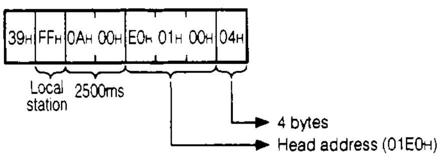
Set to "00H" when specifying the byte length as 256 bytes.

Response format

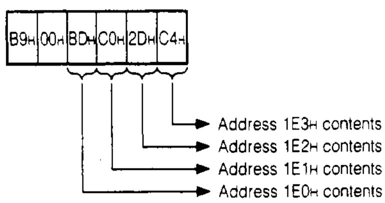


Example: When reading the comment (1E0H to 1E3H) of the PLC CPU installed in the E71.

Command (remote node → E71)

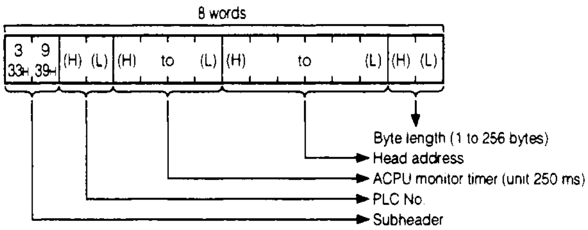


Response (E71 → remote node)



(b) When exchanging using ASCII code

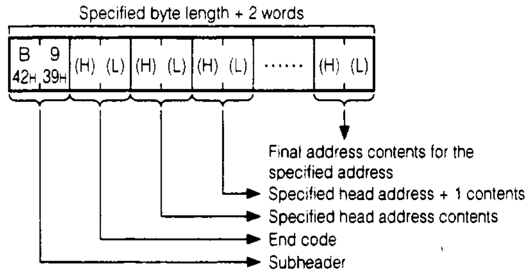
Command format



Remarks

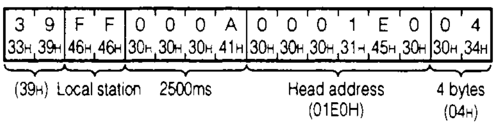
Set to "3030H" when specifying the byte length as 256 bytes.

Response format

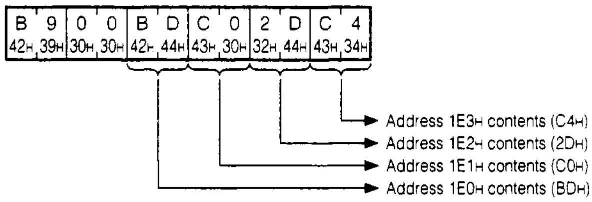


Example: When reading the comment (1E0H to 1E3H) of the PLC CPU installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)

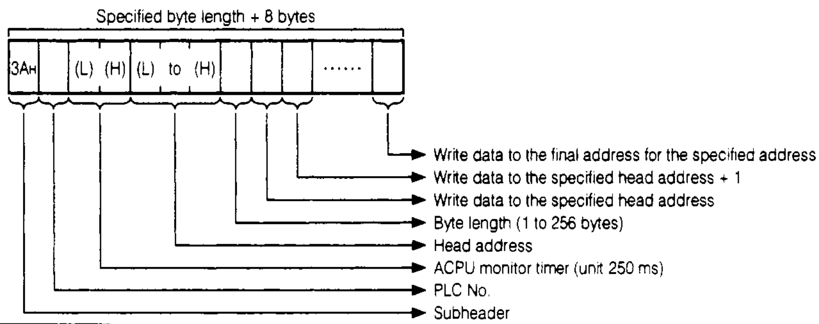


3 Batch write

This section explains the command/response format when batch writing to the comment memory.

(a) When exchanging using binary code

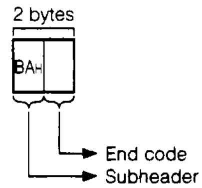
Command format



Remarks

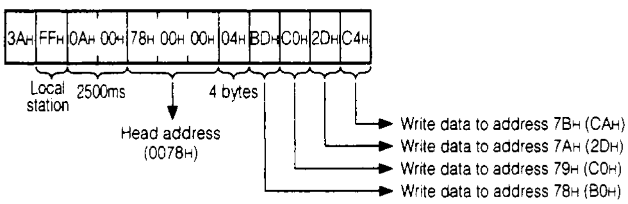
Set to "00_H" when specifying the byte length as 256 bytes.

Response format

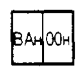


Example: When writing data to the comment area (78_H to $7B_H$) of the PLC CPU installed in the E71.

Command (remote node → E71)

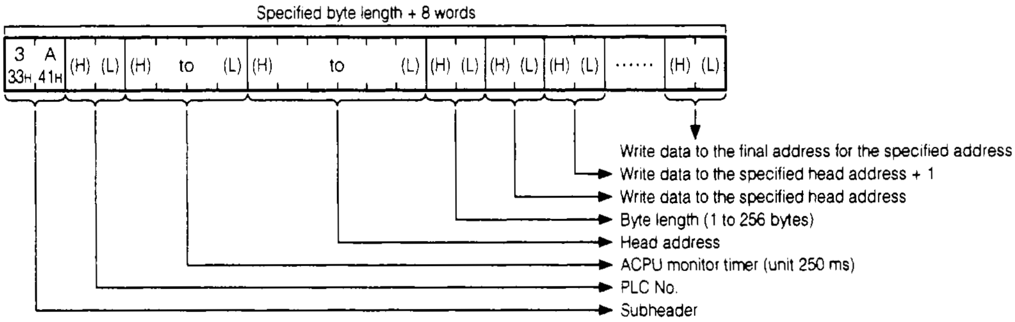


Response (E71 → remote node)



(b) When exchanging using ASCII code

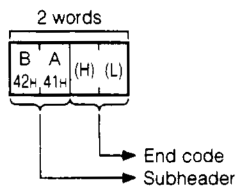
Command format



Remarks

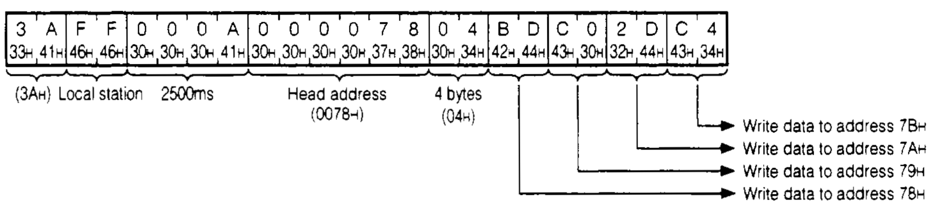
Set to "3030H" when specifying the byte length as 256 bytes.

Response format

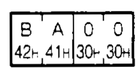


Example: When writing to the comment area (78H to 7BH) of the PLC CPU installed in the E71.

Command (remote node → E71)



Response (E71 → remote node)



10.7 Loopback Test

The loopback test is a function that tests whether or not the exchange between a remote node and the E71 is normal. The data transmitted from a remote node is returned as a response unchanged to the transmission origination station from the E71.

(1) The function used for the loopback test are shown in Table 10.14.

Table 10.14 Functions List

Item	Command/ response classification	Processing description	Number of processing points conducted in one exchange	PLC CPU status		
				Stopped	Running	
					Write possible setting	Write impossible setting
Loopback test	16H	The characters received from the remote node are returned unchanged to the remote node.	256 bytes	○	○	○

Point

For the transmission/reception text during the loopback test, for the transmission data portion transmit the following data as the header portion.

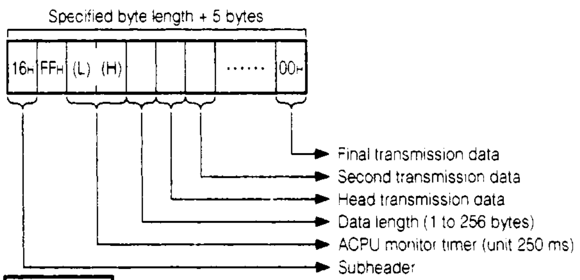
① When exchanging using binary code
The maximum 256 byte portion numerical (00H to FFH) data.

② When exchanging using ASCII code
The maximum 256 character portion half width characters ("0" to "9," "A" to "F") data.

This section explains the command/response format when conducting a loopback test.

(a) When exchanging using binary code

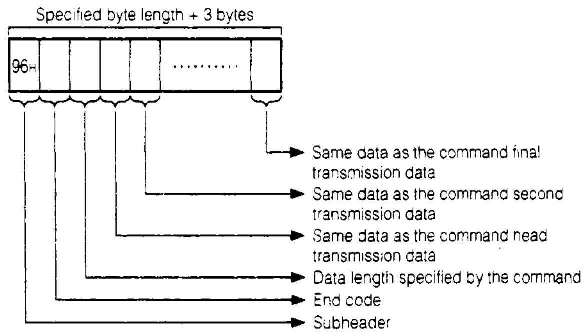
Command format



Remarks

Set to "00H" when specifying the data length as 256 bytes.

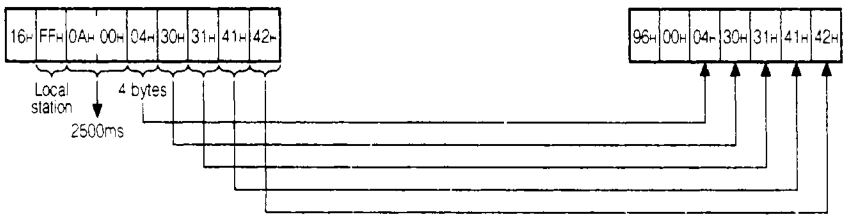
Response format



Example: When conducting a loopback test using data (30H, 31H, 41H, and 42H).

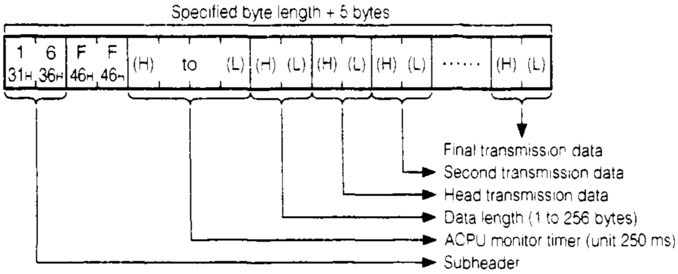
Command (remote node → E71)

Response (E71 → remote node)



(b) When exchanging using ASCII code

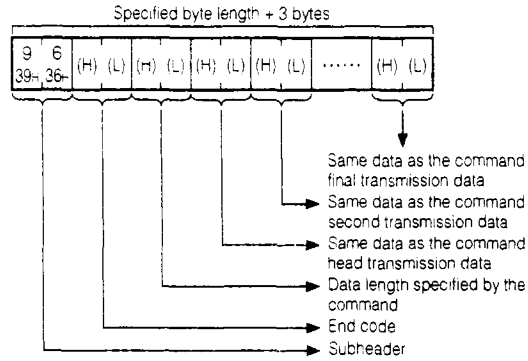
Command format



Remarks

Set to "3030H" when specifying the data length as 256 bytes

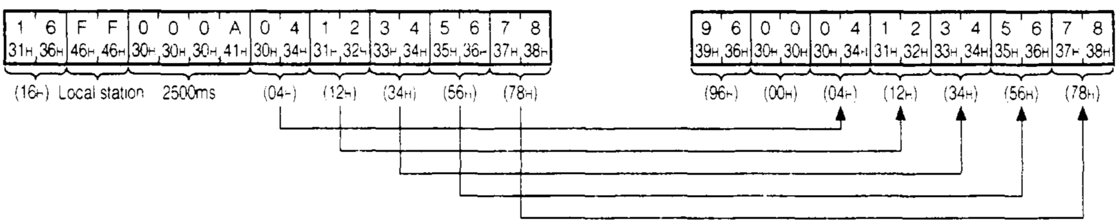
Response format



Example: When conducting a loopback test using data ("12345678").

Command (remote node → E71)

Response (E71 → remote node)



SPECIAL FUNCTIONS SECTION

The special functions section gives a function summary and explains the usage method for the special functions used by the E71 by dividing the section into one chapter per function.

The user only needs to read the chapter that explains the function to be used.

11. WHEN SETTING A SUBNET MASK

When multiple nodes are connected to one network and that network is divided and managed as virtual multiple subnetworks, then subnet mask must be created for the affected nodes. This chapter explains how to set subnet masks.

11.1 Subnet Mask

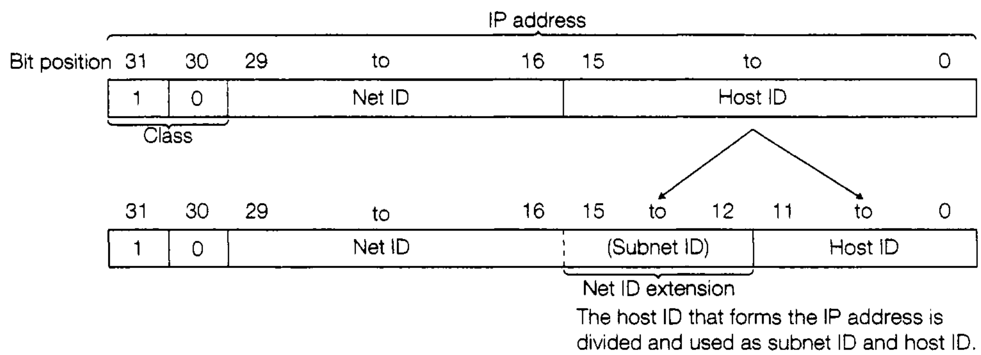
Networks build with Ethernet consist of small scale network systems where multiple nodes are connected to one Ethernet and medium and large size network systems where these smaller networks are connected using multiple routers.

The IP address of nodes connected to the Ethernet show the nodes address on that network, so three classes from class A through class C are provided to make it possible to select the address system that meets the needs of a particular network size, and the corresponding IP address is expressed using 32-bit numerals. (Refer to Item 11.3)

Subnet masks make it easy to theoretically divide one network that has many nodes connected to it into multiple subnetwork units that are easy to manage.

In particular this is information that uses a part of the host ID as a net ID extension as recognized it as the subnet ID shown below.

(Example using class B)



Point

(1) All nodes on the same subnetwork must have a common subnet mask.
Refer to Item 12.3 "Summary of Router Relay Processing" for details on the transmission from E71 when the net ID (including subnet ID) of the data transmission destination node differs from the local station.

(2) If not managed as a subnetwork then each node does not have to have a subnet mask.

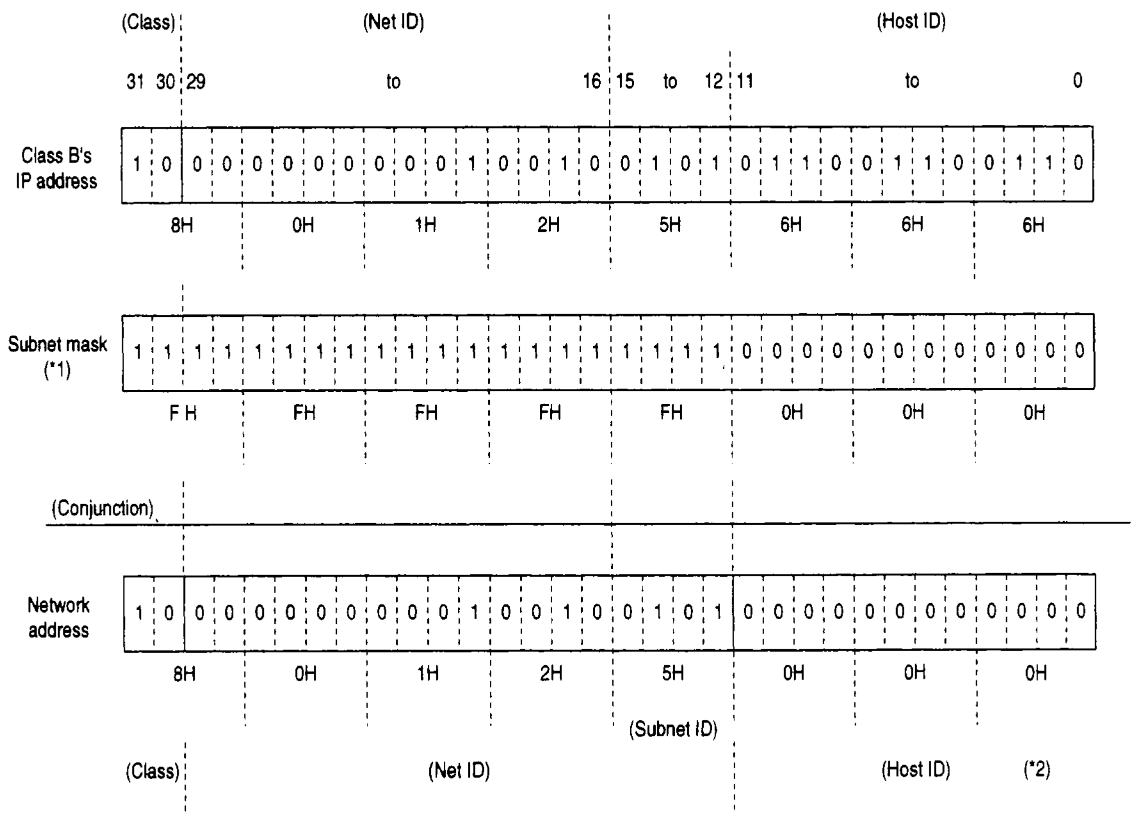
For the E71 the subnet mask is set using the buffer memory subnet mask setting area (Subnet mask field) and is specified how far a host ID like that below is extended.

① The location up to where one subnet mask field is created becomes the net ID + subnet ID (extension net ID) and the E71 handles this portion as a net ID.

② If the subnet IDs differ, they will be viewed as separate networks.

* When Ethernets are connected using routers, specifying a subnet ID makes it possible to see which router is managing which network.

(Example) When FFFFF000H is used as the class B subnet mask



*1 The subnet mask sets up bit masked value where the net ID portion and subnet ID portion are "1," and the host ID portion is "0."

*2 001H to FFEH can be used as the host ID.

11.2 Data for Setting the Subnet Mask

This section explains the subnet mask setting area which is used to divide the network to multiple virtual subnetworks to make management of one network easier.

The network manager (the person who plans the network or manages the IP addresses, etc.) needs to set the values for creating this area before initial processing.

Buffer Memory

(Address)	Subnet Mask Setting Area	Default Value
1C0 to 1C1H (448 to 449)	Subnet mask field (2 Words)	0H (0)

1 Subnet mask field (default value = 0H) address 1C0H to 1C1H (448 to 449)

(a) Sets the field values used to find the subnet address.

(b) Specify the setting values as shown below.

① When using a subnet, specify C0000000H to FFFFFFFCH.

② When not using a subnet, specify the local station's net ID net mask.

This will make it so that the subnet ID and the net ID are handled as the same thing.

- Specify the values in the following table for subnet masks when a subnet is not used.

Class	Mask Value
Class A	FF000000H
Class B	FFFF0000H
Class C	FFFFFF00H

(c) If a subnet mask field value where all of the local station net IDs cannot be masked is specified, all of the net IDs will automatically be given masked values.

(Example) When the Local Station is Class B

Following shows an example specification to the subnet mask field and the corresponding actual subnet mask's value. (Refer to Item 11.3 for the allocation of class B IP address)

Specified value to the subnet mask field	Actual subnet mask value
FF000000H	FFFF0000H
FF008000H	FFFF8000H
FFFF8000H	FFFF8000H

(To mask all net IDs subnet mask field of FFFF0000H or higher is required.)

(d) When setting subnet masks, perform the following settings besides the setting for the subnet mask setting area:

- "Use router relay function" setting in the special function settings (address 2)
- Settings for the router relay function (address 450 to 472)

11.3 Ethernet IP Address

This section explains in summary the Ethernet IP address and the classes, net ID, and host ID it contains.

1 Nodes connected to the Ethernet have both a fixed Ethernet address and a voluntary IP address for just that node.

Users do not need to be aware of the Ethernet address because it is handled by the ARP (Address Resolution Protocol).

The IP address shows the address on the network for each node connected to the Ethernet, so the user must remember this ID.

The IP address is divided into three classes of class A through class C to allow selection of the address system that is most suitable for the size of the network system.

A network manager (the person who plans the network, manages IP addresses, etc.) must set a 32-bit numerical value for each node following the standard IP address method used on a world wide scale.

2 The contents and role of each class, network ID, and host ID are as follows.

① Class A to Class C

Class A is for networks with many hosts, class C is for networks with few hosts, and class B is for networks of intermediate size. (Up to 254 hosts can be connected with class C.)

② Net ID

This is used to identify the network to make it possible to handle multiple networks. Networks with different network IDs will be identified as separate networks.

③ Host ID

This ID is used to identify hosts on a network.

When using the subnet mask described in this chapter, the host ID portion can be eliminated and an extension added to the net ID.

- If the IP address is compared to a telephone number, the network ID, host ID, and port No. specified during data exchange would play the following roles.

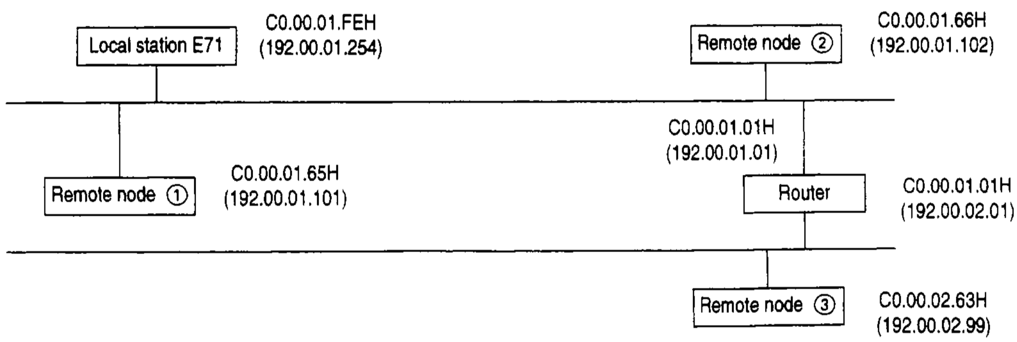
  Net ID : long distance number, Host ID : telephone number,
  Port No. : extension number

  If the network IDs (long distance numbers) are the same then direct exchange is possible, but if they are different then exchange must be conducted by going through routers (telephone exchanges).

  The function used for exchanging via routers (telephone exchanges) is the router relay function described in Chapter 12.

  (Example) When the local station E71's IP address is class C (Refer to 3)

  (IP address top stage: hexadecimal expression, bottom stage: decimal expression)



3 **Shows the IP address allocation for each class.**

When there are many nodes in the same network, then there must be many host IDs.

In addition, when there are few nodes in one network, but there are many networks, then there must be many network IDs.

(a) Class A, class B, and class C are identified by the first 2 bits of the IP address.

(b) In the class A IP address, the net ID is allocated to the 7th bit and the host ID is allocated to the 24th bit.

	31	30 to 24	23 to 0
Class A	0	Net ID	Host ID

(c) In the class B IP address, the net ID is allocated to the 14th bit and the host ID is allocated to the 16th bit.

	31	30	29 to 16	15 to 0
Class B	1	0	Net ID	Host ID

(d) In the class C IP address, the net ID is allocated to the 21st bit and the host ID is allocated to the 8th bit.

	31	30	29	28 to 8	7 to 0
Class C	1	1	0	Net ID	Host ID

Point

Limitations when setting E71's IP address are shown below. There are no net ID limitations.

(1) Set the class to class A to C.

(2) For the host ID (*1), make it so that all of the host ID range bits are not 0/1.

*1 When using the router relay function and the subnet mask, the host ID masked the net mask will become the subject host ID.

(3) Use IP addresses that do not overlap with remote node including those on other networks.

Remarks

Net addresses

A net address is an IP address for which a host ID is 0.

	31	30	29	28 to 24	23 to 16	15 to 8	7 to 0
Class A	0	Net ID			0		
Class B	1	0	Net ID			0	
Class C	1	1	0	Net ID			0

MEMO

12. WHEN USING ROUTER RELAY FUNCTIONS

This section explains when a PLC CPU and a remote node are exchanging via a specified router relay.

12.1 Router Relay Functions

Normally, in an Ethernet that uses IP, exchange can only be conducted between the local station and remote nodes connected to the same Ethernet (have the same net ID).

To communicate with a remote node on a different Ethernet (different net ID), a router relay must be used.

This router relay function is a static router relay function used for exchange with the remote node of a different Ethernet (Different net ID).

Using this function makes it possible to exchange through routers and gateways using PLC CPU side TCP/IP active open and UDP/IP transmission.

(The router relay function does not operate as a router.)

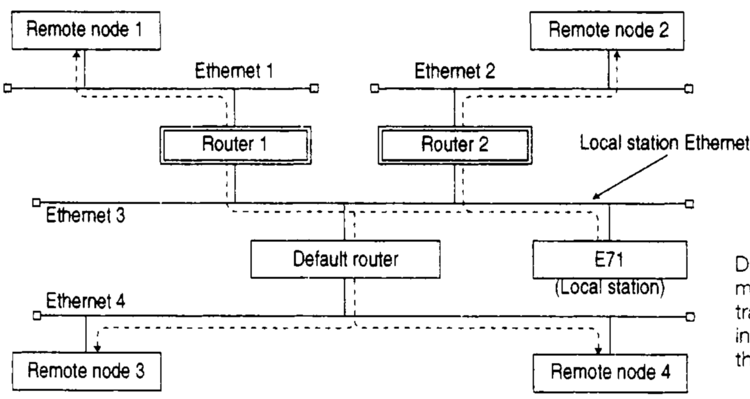
Exchange using the E71's router relay function is conducted when the buffer memory is given the following settings by the user initial processing.

- Special function setting: .... sets "using router relay function"
- Routing information settings: .... set a default router and a maximum of 5 voluntary routers

When exchanging data, the E71 conducts router relay exchange with another Ethernet using the following method when the partner station subnet ID in the message differs from the local station subnet ID.

① The subject router relays the exchange when there is a partner station subnet address in the user set routing information.

② Relay exchange is conducted using the default router when there is no partner station subnet address in the user set routing information.



During transmission, the E71 does not transmit directly to the partner station, but instead transmits the data once to the router specified in the routing information. The router transmits the received data to the partner station.

Point

(1) It is not necessary to use the router relay function when the E71 is exchanging with a partner remote node using a router relay in passive open (TCP/IP).
In this case, exchange can be done even if the router relay function is not used.

(2) The router relay function is not required in systems that use proxy routers.

12.2 Exchangeable Functions and Settable Range Using Router Relay Functions

This section explains the data exchange functions and settable range used to exchange data with remote Ethernets using router relay functions.

1 Exchangeable functions

When conducting the following data exchange, exchange with remote Ethernets can be done using the router relay function.

- Fixed buffer exchange (possible for either with procedure and without procedure)
- Random access buffer exchange
- Reading/writing data to the PLC CPU

2 Settable range

This makes it possible to exchange with remote nodes connected to Ethernet within the range accessible via the router. There is one default router and 5 voluntary routers set in the E71 through which exchange can be conducted.

12.3 Summary of Router Relay Processing

This section gives an overview of the router relay processing conducted by the E71 when data is transmitted to remote nodes on remote Ethernets via routers.

The E71 determines the transmission destination of the data at the time data is transmitted using the following procedure.

1 Checks whether there is a setting that uses the router relay function.

If the setting does not use the router relay function, exchange is done directly with the partner remote node.

2 Checks whether the partner remote node is connected to the same Ethernet as the local station by using the following two formulas. (Subnet ID check)

If they are the same, then it is determined that it is the same Ethernet and exchange is done directly with the partner remote node. If they are different, it is judged that it is a remote Ethernet, and exchange to the partner node is conducted via a router.

(Formula-1) Partner station IP address and subnet mask file conjunction
(Formula-2) Local station ID address and subnet mask field conjunction

3 When it is recognized that the partner remote node is connected to a remote Ethernet, exchange with the partner remote node is conducted via a router.

(a) When the partner remote node is the same class as the local station

Compares the above (formula-1) with the valid subnet addresses 1 to 5 in the routing information.
If the settings are the same, exchange is conducted to the router supporting the router IP address.
If the settings are different, exchange is conducted to the default router.

(b) When the partner remote node and the local station are different classes

The partner remote node's net address and the valid subnet addresses 1 to 5 in the routing information are compared. If they are the same, then exchange is conducted to the router that supports the router IP address. If they are different, then exchange is conducted to the default router.

12.4 Data for Using Router Relay Functions

This section explains about the parameter setting area used to conduct exchange with remote nodes using the router relay function.

The network manager (the person who plans a network, manages IP addresses, etc.) sets the values for the area before initial processing.

Buffer Memory

(Address)	Routing Information (36 Words)			Default Value
1C2 to 1C3H (450 to 451)	Default router IP address		(2 Words)	0H (0)
1C4H (452)	Registered number of routers		(1 Word)	0H (0)
1C5 to 1C6H (453 to 454)	Router 1 setting	Subnet address 1	(2 Words)	0H (0)
1C7 to 1C8H (455 to 456)		Router IP address 1	(2 Words)	0H (0)
1C9 to 1CAH (457 to 458)	Router 2 setting	Subnet address 2	(2 Words)	0H (0)
1CB to 1CCH (459 to 460)		Router IP address 2	(2 Words)	0H (0)
1CD to 1CEH (461 to 462)	Router 3 setting	Subnet address 3	(2 Words)	0H (0)
1CF to 1D0H (463 to 464)		Router IP address 3	(2 Words)	0H (0)
1D1 to 1D2H (465 to 466)	Router 4 setting	Subnet address 4	(2 Words)	0H (0)
1D3 to 1D4H (467 to 468)		Router IP address 4	(2 Words)	0H (0)
1D5 to 1D6H (469 to 470)	Router 5 setting	Subnet address 5	(2 Words)	0H (0)
1D7 to 1D8H (471 to 472)		Router IP address 5	(2 Words)	0H (0)

1 **Default router IP address (Default value = 0H) ...... Address 1C2H to 1C3H (450 to 451)**

(a) Sets the router (default router) IP address to which exchange will be conducted when exchange with a remote node on a remote Ethernet is done via other than the router to be specified using the following subnet address n and router IP address n.

(b) Sets the settings value to a value other than 00000000H and FFFFFFFFH.

(c) Because the default router itself must be on the same network as the local station, the default router's subnet ID must be the same as the local station's subnet ID.

2 **Number of registered routers (Default value = 0H) .... Address 1C4H (452)**

(a) Sets the number of subject routers (number of valid settings) using the following subnet address n and router IP address n when conducting exchange with a remote node on a remote Ethernet via anything other than the default router.

(b) Sets the setting value to 0 to 5. (If a value higher than 6 is set, it will be seemed as 5.)

(c) Sets the specified numbers portion of the subnet address n and router IP address n in the following 3 and 4 area.

3 Subnet address n (Default value = 0H) .... Address 1C5H....(453...)

(a) Sets the partner station's subnet address and net address when conducting exchange with a remote node in a remote Ethernet via anything other than the default router.

(b) Sets to either of the following when the settings value is other than 00000000H and FFFFFFFFH.

① Specify the partner station's subnet address when the partner station and the local station are the same class.

(Example) The subnet address is as follows when the partner station's IP address is 59010201H.

- 59000000H when the subnet mask is specified as FF000000H.
- 59010200H when the subnet mask is specified as FFFFFF00H.

② When the partner station and the local station are different classes, specify the partner station's net address.

4 Router IP address n (Default value = 0H) .... Address 1C7H....(455...)

(a) Specify the IP address of the router to which exchange will be conducted when conducting exchange with a remote node on a remote Ethernet via anything other than the default router.

(b) Because the router itself and the local station must be on the same network, the router's subnet ID and the local station's subnet ID must be the same.

Point

When using the router relay function set the following settings in addition to settings in the routing information area.

* Set "using router relay function" (address 2H(2)) in the initial processing parameter setting area's special function setting area.

TROUBLESHOOTING SECTION

The troubleshooting edition explains about the error codes corresponding to errors, error contents, error processing, and troubleshooting flow when trouble occurs during Ethernet interface module use.

13. TROUBLESHOOTING

This section explains about trouble that occurs when using the Ethernet interface module and covers error codes, error description, error processing, and trouble shooting flow for errors detected by the E71.

When trouble occurs that prevents normal exchange between the E71 and a remote node then the problem must be limited to whether the cause occurred on the E71 end, in the line, or at the remote node end.

When the trouble occurs at the E71 end, conduct trouble processing by using the error codes stored in the buffer memory exchange state storage area (Refer to Item 5.5.2) and the error log area (Refer to Item 5.5.3).

Remarks

When a line error, etc., occurs when equipment from different manufacturers is connected, we ask that the user use a line analyzer, etc., to determine the location of the problem.

13.1 List of Error Codes

This section explains about the error codes (End code, Error code), error description, and error processing that are generated for each process when data is exchanged between the E71 and a remote node.

The types of errors that occur are shown below.

<table>
<tr><th></th><th>Error Type</th><th colspan="2">Description</th><th>Error Code Storage Buffer Memory Address</th><th>Reference Item</th></tr>
<tr><td>1</td><td>Errors that occur during Initial processing</td><td colspan="2">• Setting value error
• Initial processing error</td><td>50H (80)</td><td rowspan="5">Item 13.1.1</td></tr>
<tr><td>2</td><td>Errors that occur during open processing</td><td colspan="2">• Setting value error
• Open processing error</td><td>5DH (93) ...
A9H (169) ...</td></tr>
<tr><td>3</td><td>Errors that occur during fixed buffer transmission</td><td colspan="2">• Specified data error
• Transmission error</td><td>5EH (94) ...
5FH (95) ...</td></tr>
<tr><td>4</td><td>Errors that occur during field buffer exchange</td><td colspan="2">• Specified data error
• Exchange error (excluding 3 above)</td><td>5FH (95) ...</td></tr>
<tr><td>5</td><td>Errors that occur during exchange other than the above
* Errors for which error codes are stored in the error log area</td><td colspan="2">• Specified data error
• Errors where the place of origin of the error cannot be determined
• Errors occurring during random access buffer exchange, reading/writing data to the PLC CPU</td><td>A9H (169) ...</td></tr>
<tr><td rowspan="5">6</td><td rowspan="5">Data exchange errors
* Errors return to the request originating remote node</td><td colspan="2">• Errors returned by fixed buffer exchange (End code)</td><td rowspan="5">—</td><td rowspan="2">Item 13.1.1</td></tr>
<tr><td colspan="2">• Errors returned by random access buffer exchange (End code)</td></tr>
<tr><td colspan="2">• Errors returned by reading/writing data to the PLC CPU</td><td>—</td></tr>
<tr><td rowspan="2"></td><td>End code during E71 command use</td><td>Item 13.1.1</td></tr>
<tr><td>Error code during E71 command use</td><td>Item 13.1.2</td></tr>
</table>

13.1.1 End Codes Returned to the Remote Node during Data Exchange Error Codes Stored in the Buffer Memory

This section explains the error contents and error processing for end codes return to remote node and error codes stored in the E71's buffer memory when an error occurs during processing for data exchange between the E71 and a remote node.

<table>
<tr><th>End Code
Error Code</th><th>Description</th><th>Processing</th></tr>
<tr><td>00H</td><td>Normal end</td><td>——</td></tr>
<tr><td>01H</td><td>• The exchange data length exceeds the set range.</td><td>Correct the transmission data data length. (Refer to Item 3.3.)</td></tr>
<tr><td>50H</td><td>• When the codes are other than those prescribed by the subheader commands and responses.
<table>
<tr><th>Exchange Processing</th><th>Commands/
Responses</th></tr>
<tr><td>Fixed buffer exchange</td><td>60H</td></tr>
<tr><td>Random access buffer exchange</td><td>61H, 62H</td></tr>
<tr><td>Reading and writing data
in the PLC CPU</td><td>00H, 3CH</td></tr>
</table>
• During fixed buffer exchange, when the actual data quantity is less than the data length setting, the remaining data is determined to be second data and processed. In this case, a subheader command undefined error will occur.</td><td>• Check in correction of the set commands and responses at the remote node.
(The E71 automatically adds the commands and responses, so the user does not need to set these.)
• Refer to the remarks at the end of this page.
• Check and correct the data length.</td></tr>
<tr><td>51H</td><td>• For random access buffer exchange, the specified header address from the remote node is set outside the range of 0 to 6143.</td><td>• Check and correct the specified header address.</td></tr>
<tr><td>52H</td><td>• For random access buffer exchange, the specified header address from the remote node + number of data words (set during read) exceeds the range of 0 to 6143.
• The data (text) for the specified number of words cannot be transmitted in one frame. (The transmission/reception data length value or text amount is not within the acceptable range.)</td><td>• Check and correct the header address and number of data words.(Refer to Item 3.3)
• Correct the number of read/write points.</td></tr>
<tr><td>54H</td><td>• When the data code setting using the dip switch (SW2) on the front of the E71 is set to ASCII code, ASCII code data that cannot be converted to parity code by the remote node was transmitted.</td><td>• Check and correct the remote node transmission data.</td></tr>
<tr><td>55H</td><td>• When the CPU exchange timing setting is set to write not possible during RUN using the dip switch (SW7/SW3) on the front of the E71, and the data write request from the remote node was made while the PLC CPU was running.
• A request from a remote node to write a parameter, sequence program, or microcomputer program was received while the PLC CPU was running. (Not related to the on/off state of the dip switch (SW7/SW3) on the front of the E71.)</td><td>• Conduct data write while the SW7/SW3 is on (write allowed during RUN). However, parameters, sequence programs, and microcomputer programs cannot be written while the CPU is running.
• Write the data after stopping the PLC CPU.</td></tr>
<tr><td>56H</td><td>• There is a device setting error from a remote node.</td><td>• Correct the device setting.</td></tr>
</table>

End Code Error Code	Description	Processing
57H	• The number of command points specified by the remote node exceed that of the maximum processing number of points for each process (number of points processed during one exchange). • The header address (header address No., header step No.) to the specified number of points, exceeds the maximum address for each process (device No., step No.).	• Correct the specified number of points and the header address (device No., step No.).
	• The command byte length is longer than that prescribed. • During data write, the specified write data number of points is different from the number of points specified value.	• Check and reset the command data.
	• There was a monitor request even though monitor data is not registered.	• Conduct monitor data registration.
	• When a microcomputer program was read/written it was set beyond the parameter setting range's final address.	• Reading and writing cannot be performed after the final address. Correct the specified address.
	• During the extension file register block No. specification, a range of block No. were specified that exceed the subject memory cassette's capacity.	• Correct the block No.
58H	• The command head address from a remote node (head device No., address step No.) from a remote node that exceeds the specifiable range was set. • A microcomputer program for file register (R) read/write that is outside the PLC CPU's parameter settings was specified.	• Correct all processing to values in the specifiable range.
	• The extension file register block No. specifications are for blocks that do not exist.	• Correct the block No.
	• A file register (R) is set for the A1(N) CPU.	• The A1(N) CPU cannot use file registers.
	• A word device is specified using bit device commands. • The bit device's header No. is specified at values other than multiples of 16 using word device commands.	• Correct the commands or the specified device.
59H	• An extension file register read/write request was made to the A1(N) CPU.	• The A1(N) CPU cannot use extension file registers.
5BH	• Exchange cannot be done between the PLC CPU and the E71. • The PLC CPU cannot process requests from remote nodes.	• Prepare the error location by adding an error code (Refer to Item 13.1.2) after the end code.
60H	• The exchange time between the E71 and the PLC CPU exceeds the ACPU monitoring timer value.	• Lengthen the ACPU monitoring timer value.
62H	• Returns other than "00H" (normal end) to the response end code from a remote node for fixed buffer transmission.	• Read and process the response end code (Buffer memory address 95, 105...) from the remote node for the fixed buffer transmission.

End Code Error Code	Description	Processing
70_H	A recurring signal for the response does not arrive within the response monitor timer value. • All the data cannot be received by the allocation reception.	• Check the partner node operation. • Check if the connection cable is disconnected. • Check if there's a problem with the connection to the transceiver or with the terminator connection. • If the response monitor timer value is small then change it to a large value. • Check the transmission data on the partner node side.
71_H	• The amount of data set in the data length cannot be received within the response monitor timer value. • The actual amount of data is smaller than the value set in the date length. • The remainder of the text allocated by the TCP/UDP level was not received within the response monitor timer value.	• Correct the data length of the exchange data. • When connecting TCP exchange the interference of packets in the line can be suspected, so change the initial processing setting data. • When exchanging with UDP then conduct a retry of the transmission side program.
80_H	The corresponding connection open processing is not completed.	Conduct open processing.
81_H	An Ethernet address that does not exist is specified. (Only when the UDP I/P was used as a communication method.)	• Check the Ethernet address of the remote node with which exchange is being done. • For an ARP function then set the Ethernet address to 0_H/$FFFFFFFFFFFF_H$ and conduct initial processing.
101_H	There is an error with the E71 port No.	Correct the E71 port No. (Refer to Item 5.4.1 $\boxed{2}$ (c) ①.)
102_H	There is an error with the remote node port No.	Correct the remote node port No. (Refer to Item 5.4.1 $\boxed{2}$ (c) ③.)
103_H	There is an error with the port specified to be opened for exchange using TCP/IP.	• Correct the exchange address setting area for each connection. • Did not specify the port that is open.
104_H	Multiple connections are set with the E71 port No. when exchanging using UDP/IP.	Correct the exchange address setting area for each connection. (Refer to Item 5.4.1. $\boxed{2}$ (c) ①.)
105_H	The E71 initial processing is not finished.	Conduct E71 initial processing.
106_H	The remote node IP address was set to 0 or $FFFFFFFF_H$.	Set the remote node IP address to 1 to $FFFFFFFE_H$.
107_H	Open processing has already been conducted for the pairing open connection (For the next connection).	• Check if open processing has not been done for either of the connections for the pairing open. • Change the pairing open combination. (Refer to Item 5.4.1 $\boxed{1}$ (b) ③.)
108_H	An existence check for the partner remote node could not be done within the response monitor timer value.	• Check the operation of the partner remote node. • Review and change the set values for existence check. (Refer to Item 5.3.1 $\boxed{4}$ to $\boxed{6}$.) • Check if the connection cable is loose. • Check if there's a problem with the connection to the transceiver or with the terminal connection.
109_H	There is a set value outside the allowable range in the timer set values during initial processing.	Review and correct the timer set values during initial processing. (Refer to Item 5.3.1. $\boxed{4}$ to $\boxed{12}$.)
201_H	• There is an error in the E71 IP address set value during initial processing. • There is an error in the subnet mask field set value when using the router relay function.	• Correct the IP address. Make the class A, B, or C. (Refer to Item 5.3.1 $\boxed{1}$, and Item 11.3.) • Correct the subnet mask. (Refer to Item 11.2.)
301_H	There is an error in the subnet mask field set value.	Correct the subnet mask and reconduct initial processing. (Refer to Item 11.2.)

End Code Error Code	Description	Processing
302H	• There is an error in the router relay function default router IP address set value. • The default router IP address net ID (Net ID after subnet mask) defers from the local station E71 IP address net ID.	• Correct the default router IP address and reconduct initial processing. (Refer to Item 12.4 [1].) • Make it the same as the local station E71 IP address net ID. (Refer to Item 11.2.)
303H	There is an error in the router relay function subnet address set value.	Correct the subnet address and reconduct initial processing. (Refer to Item 12.4 [3].)
304H	• There is an error in the router relay function router IP address set value. • The router IP address net ID (Net ID after subnet mask) defers from the local station E71 IP address net ID.	• Correct the router IP address and reconduct initial processing. (Refer to Item 12.4 [4].) • Make it the same as the local station E71 IP address net ID. (Refer to Item 11.2.)
4000H to 4FFFH	An error occurred at QnACPU.	Refer to the manual for the QnACPU of the accessed station and take corrective action.
7004H	A connection is not established during TCP connection open processing.	• Confirm the operation of the other partner node. • Confirm the open processing of the other partner node. • Correct the set value for the usage of the communication parameter. (See (1) in Section 5.4.1.) • Review the port number of the E71 as well as the IP address/port number and the open method of other nodes. • Check to see if the connection cable is securely connected. • Check to see if the transceiver and terminator are correctly connected.
7010H	A transmission error has occurred.	Same as the corrective action for error code B000H. See the corrective action for B000H.
8001H	A remote node Ethernet address (Buffer memory address 28 to 30, ..., 77 to 79) of 20 digits or more outside the default (FFFFFFFFFFFFH) has been registered.	Be sure to use a default value when using an ARP function.
8002H	End processing has not been conducted.	Conduct initial processing after conducting end processing.
8003H	The next transmission request was conducted even though the transmission processing has not been completed.	Conduct the next transmission request after the transmission end signal turns on.
8004H	A system error has occurred.	Conduct E71 initial processing.
8005H	An initialization error has occurred because of an error in the initial processing parameter.	Correct the initial processing parameter setting values.
9001H	The open processing for the corresponding connection has not been completed.	Conduct open processing,
9002H	A fixed buffer transmission has been conducted during closed processing (Before the open end signal (X10 to 17) is turned off) because the open request signal (Y8 to F) turned off.	Execute so that the corresponding connection open request signal (Y8 to F) and the opened end signal (X10 to 17) are on at the same time for fixed buffer transmission.
9005H	Insufficient internal resources for the TCP transmission request. Insufficient transmission buffer.	• Retransmit the same data. • It is possible that the next transmission is conducted without waiting for a response. Conduct the next transmission after receiving a response.
9006H	Reception data check some error when TCP/IP is used.	Check the partner remote node check sum calculation.
9008H	Insufficient internal resources for the UDP transmission request. Insufficient transmission buffer.	• Retransmit the same data. • It is possible that the next transmission was conducted without receiving a response. Conduct the next transmission after a response is received.
9009H	Reception data check some error when using UDP/IP.	Check the partner remote node check sum calculation.

End Code Error Code	Description	Processing
9059_H	A TCP/ULP time out error occurs during exchange using TCP/IP. (An ACK is not returned from the remote node when using the TCP protocol.)	• Check the partner remote node operation. • Check if the connection cable is loose. • Check if there is a problem with the connection to the transceiver or with the terminator connection. • Correct the initial parameter TCP/ULP timer value.
$A001_H$ to $A004_H$	An ICMP error packet is received.	• Check if the IP address and port No. set for the E71 are correct. • Check if the partner remote node IP address and port No. are correct. • Check the partner remote node operation. • Check if the connection cable is loose. • Check if there's a problem with the connection with the transceiver or with the terminator connection.
$A006_H$	An ICMP error packet is received. (An IP assembly time out occurs at the partner remote node.)	• Check the partner remote node operation. • A packet could be in the line, so transmit after the free time has passed. • Check if the connection cable is loose. • Check if there's a problem with the connection to the transceiver or with the terminator connection. • Correct the partner remote node side IP assembly timer value when there is a time over for the existence time.
$A007_H$	An IP assembly time out error occurs. (The remaining allocation data cannot be received and a time out occurs.)	• Check the partner remote node operation. • A packet could be in the line, so transmit from the remote node after the free time has passed. • Check if the connection cable is loose. • Check if there is a problem with the connection to the transceiver or with the terminator connection. • Correct the IP assembly timer value and reconduct initial processing. (Refer to Item 5.3.1 $\boxed{11}$).
$A009_H$	The set IP address remote node does not exist.	• Review and correct the partner remote node IP address and Ethernet address. (Refer to Item 5.4.1 $\boxed{2}$(c) 2 4.) • Set the default value when there is an ARP function in the partner remote node or set the partner remote node Ethernet address when there is no ARP function. • Check the partner remote node operation. • There could be a packet in the line so transmit after the free time has passed. • Check if the connection cable is loose. • Check if there's a problem with the connection to the transceiver or with the terminator connection.
$A00B_H$ $A00C_H$	• An ICMP error packet was received. • An ICMP error packet that was not supported was received.	• Check the partner remote node operation. • There could be a packet in the line, so transmit after the free time has passed. • Check if the connection cable is loose. • Check if there's a problem with the connection to the transceiver or with the terminator connection. • Correct the partner remote node IP assembly timer value when there is a time over of the existence time. • For the current E71 only support a return signal for echo, times stamp, and information request. Make it so that any request other than these are not transmitted from the partner remote node.
$A00D_H$	There is an error in the header check sum of the received IP packet.	• Review and transmit the correct value for the check sum transmitted by the partner remote node. • Investigate the environment's state in the line. (Noise environment, distance between the line and power lines, equipment grounds)

End Code Error Code	Description	Processing
A00EH	Cannot transmit since no space is available in internal buffers such as IP header buffer.	• Transmit the same data once again and confirm the response returned.
	The number of communication destination nodes after initial processing exceeded 20 stations. (Refer to Item 5.2 Point)	• Reduce the number of external nodes to communicate. • End communication with all external nodes and perform the E71 initial processing once again.
A00FH	The number of partner remote nodes with which exchange was done exceeded 20 stations after initial processing. (Refer to Item 5.2 Point)	• Reduce the number of remote nodes with which communication is done. • Reconduct E71 initial processing after communication with all remote nodes is completed.
A010H	• Transmission was requested to a remote node for which the class net ID differs from that of the local station when the router relay function was not used. • There is an error in the routing information setting area.	• Set the router relay function to be used and conduct initial processing. • Set the correct data in the routing information area and conduct initial processing. • Correct the transmission destination remote node IP address and conduct open processing. • Check if the net ID is correct. When making a change reconduct initial processing.
A011H	There is an error in the partner remote node side IP address setting value. • Setting the IP address to FFFFFFFFH is not possible when using TCP.	Correct the IP address. (Refer to Item 5.4.1 [2] (c) ②.) • Simultaneous broadcast communication is not possible using TCP/IP.
B000H	A transmission error has occurred.	• Check the transceiver and partner remote node operation. • Use a transceiver for which the SQE test can be conducted. • There could be a packet in the line so transmit after the free time has passed. • Check if the connection cable is loose. • Check if there's a problem with the connection to the transceiver or with the terminator connection. • Conduct a loopback test. (Refer to Item 10.7) and check whether there's a problem with the line. • Conduct a self diagnostic test and check whether there's a problem with the E71.
B001H	Transmission processing could not be conducted because the cable is not connected or is loose.	

Remarks

The exchange data is sometimes divided and exchanged because of local station and partner station buffer restrictions.

The divided reception data is restored (reassembled) by the E71 and is transmitted by the fixed buffer or the random buffer. The restoration (reassembly) of the received data is done based on the data length in the exchange data.

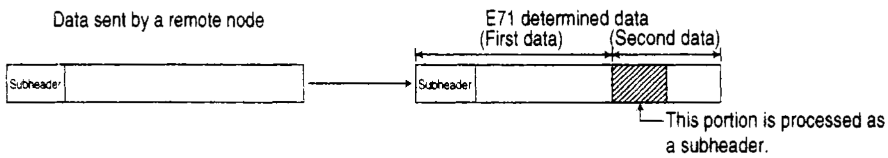
An error will occur if the exchange data setting value and actual data length are different.

1 When the set data length is smaller than the actual data amount

Wait until the data amount set in the data length is transmitted. An error will occur if the remaining data is not received within the response monitor timer, so close processing will automatically be conducted for the connection.

2 When the actual data amount is larger than the set data length

The system will attempt to process as the first data the data amount set in the data length and as the second data the next data. Because this second data does not have a subheader, a command/response undefined error will occur.



The response at this time is to return a code of 1 for the code's first bit position that was processed as a subheader.

For example, if the command subheader portion is 65H, the response subheader will become E5H.

13.1.2 Error Codes Returned to the Remote Node by Reading and Writing Data in the PLC CPU

This section explains about the error codes that are attached to the responses to the data read/writes to the PLC CPU using E71 commands. (Error codes are only attached when the end code is 5B.)

End codes (error codes) with responses attached are described in Item 13.1.1.

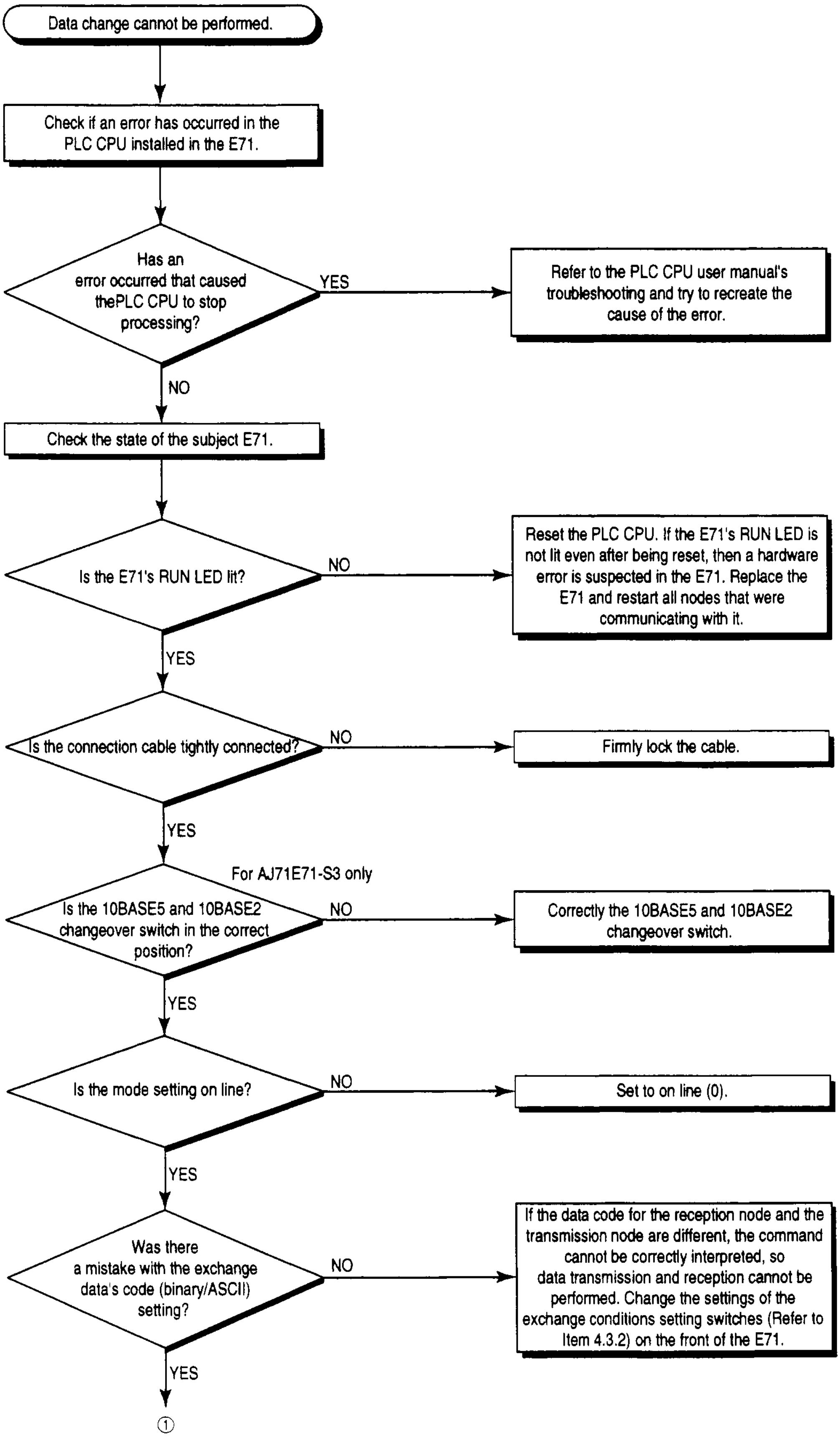
Response format	Subheader	End code	Error code	OO

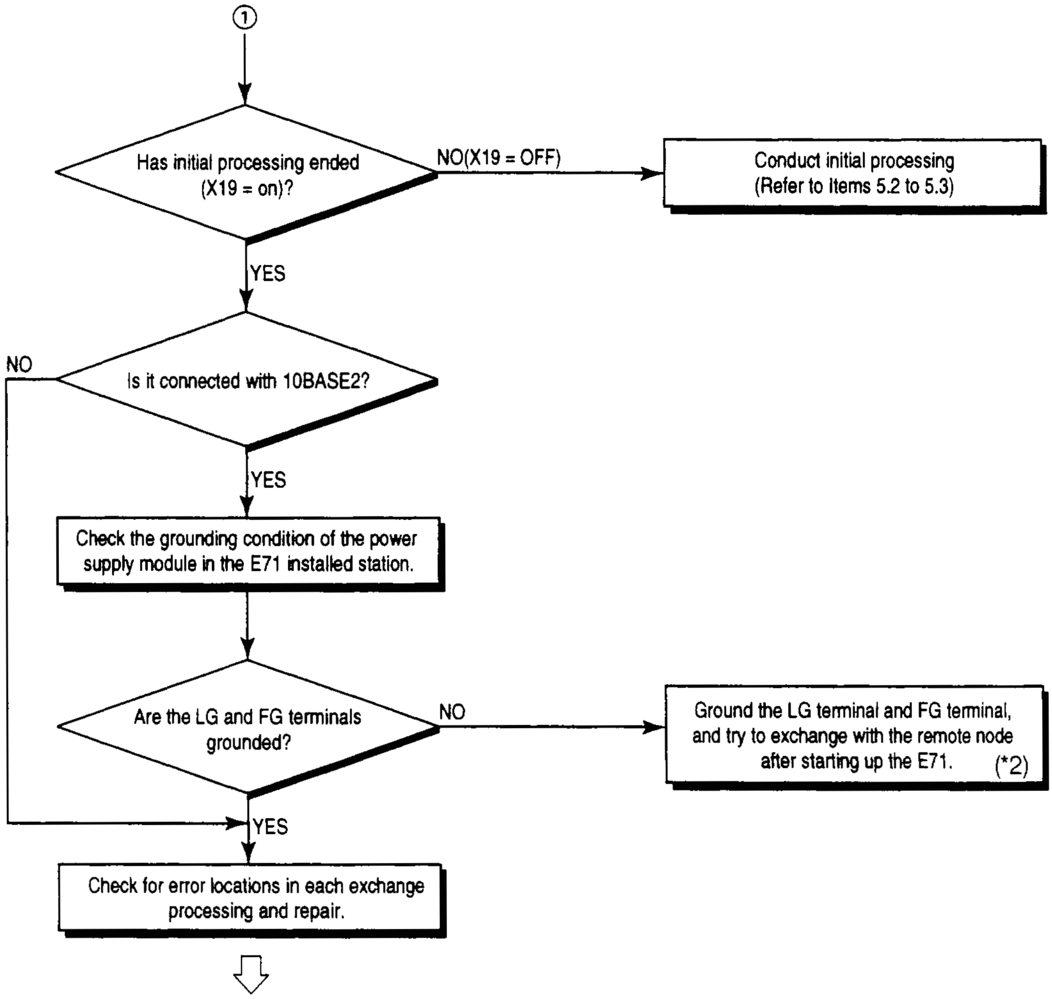
→ When an error code is stored, the end code becomes 5BH.

Error Code (Hexadecimal)	Error Item	Error Description	Processing Method
10_H	PLC No. error	The PLC No. station does not exist. The PLC No. specified by a command is other than the station No. specified for the local station "FF" and the MELSECNET link parameter settings.	Change the PLC No. to the local station's "FF" or set station No. using the link parameter and reconduct the exchange.
11_H	Mode error	Exchange defect between the E71 and PLC CPU After a request from a remote node has been received normally by the E71, for some reason (noise, etc.) normal exchange cannot be conducted between the E71 and the PLC CPU.	Reconduct the exchange. If an error occurs again, check for noise, etc., and then reconduct exchange with the E71.
12_H	Special functions unit setting error	Special Functions Unit Error There is no buffer memory with an exchangeable special functions unit in the location specified for the special functions unit No. (For example, the location has an I/O unit or an open slot in the location.).	Change the control procedure specified data contents, or change the special functions unit installation position and reconduct exchange.
13_H	Program step No. specification error	Sequence program program step No. specification error. A step No. that exceeds the program capacity range set by the PLC CPU parameter has been specified.	Set a step No. that is within the specified range or change the PLC CPU parameter contents and reconduct communication.
18_H	Remote error	Remote RUN/STOP can not be conducted. Remote STOP/PAUSE has already been conducted by another unit (another E71, etc.).	Check if remote STOP/PAUSE is working or not from other units, perform a delete, and reconduct the exchange.
20_H	Link error	The request destination CPU unit is disconnected from the data link.	Check if the PLC CPU of the station No. specified in the PLC No. is parallel off. After removing the parallel off cause, reconduct exchange.
21_H	Special functions unit bus error	The special function unit's memory cannot be accessed. (1) There is a control bus error with the special functions unit. (2) The special functions unit is damaged.	There is a hardware error in the PLC CPU, base unit, special functions unit, or E71. Consult with your nearest service center, agency, or branch office.

13.2 Troubleshooting Flow

Following is a simple troubleshooting flowchart for when exchange cannot be conducted between the E71 and the remote node. (*1)





Transmission errors during fixed buffer exchange............Item 13.2.1
Reception errors during fixed buffer exchange................ Item 13.2.2
Random access buffer exchange errors...........................Item 13.2.3
Data write/read in the PLC CPU errors...............................Item 13.2.4

*1 About turning on both the X1C of the input/output signal and display LED's COM. ERR

(1) The E71 performs the following processing when an error such as a communication error is detected.

- Stores an error code in one of the following buffer memory areas (an area that corresponds to the detected error).
- Turns on the X1C of the input/output signal (COM. ERR LED on signal).
- Turns on the display LED's COM. ERR (communication error detection display).

Area name	Address (Hexadecimal (Decimal))	Remarks
Initial error code storage area	50H(80)	——
Open error code storage area	5DH (93)	The addresses shown at left are for connection 1.
Fixed buffer transmission error code storage area	5EH (94)	
Fixed buffer communication end code storage area	5FH (95)	
Error log area	A9H (169) to B3H (179)	——

(2) When the X1C of the input/output signal is on and the display LED's COM. ERR is on, check an error code stored in one of the above buffer memory areas, verify the error content according to the description in Item 13.1.1, and take an appropriate corrective action.

(3) The Y17 of the input/output signals (COM. ERR LED off request signal) is used to turn off the X1C of the input/output signal and display LED's COM. ERR.
(Refer to Item 3.6.2(11))
After completing the following operation, turn off the X1C of the input/output signal and display LED's COM. ERR.

- Turn off all input signals (open error detection signal (X18), transmission error detection signal (X1), etc.) that are currently on by error detection.
- Check the error code and error content.

* Although it depends on the system specifications, it is recommended to turn off the X1C of the input/output signal and display LED's COM. ERR after a corrective action has been taken for the error.

*2 Be sure to ground the FG terminal and LG terminal of the power supply module in the E71 installed station. If it is not grounded correctly, you can not exchange with the remote node because the influence of noise close (disconnect) the communication line.

Read carefully the PLC CPU User's Manual that describes the procedure for the installation or wiring work. And then, turn off the E71 installed station power supply in order to reset the grounding of the LG terminal and FG terminal.

Point

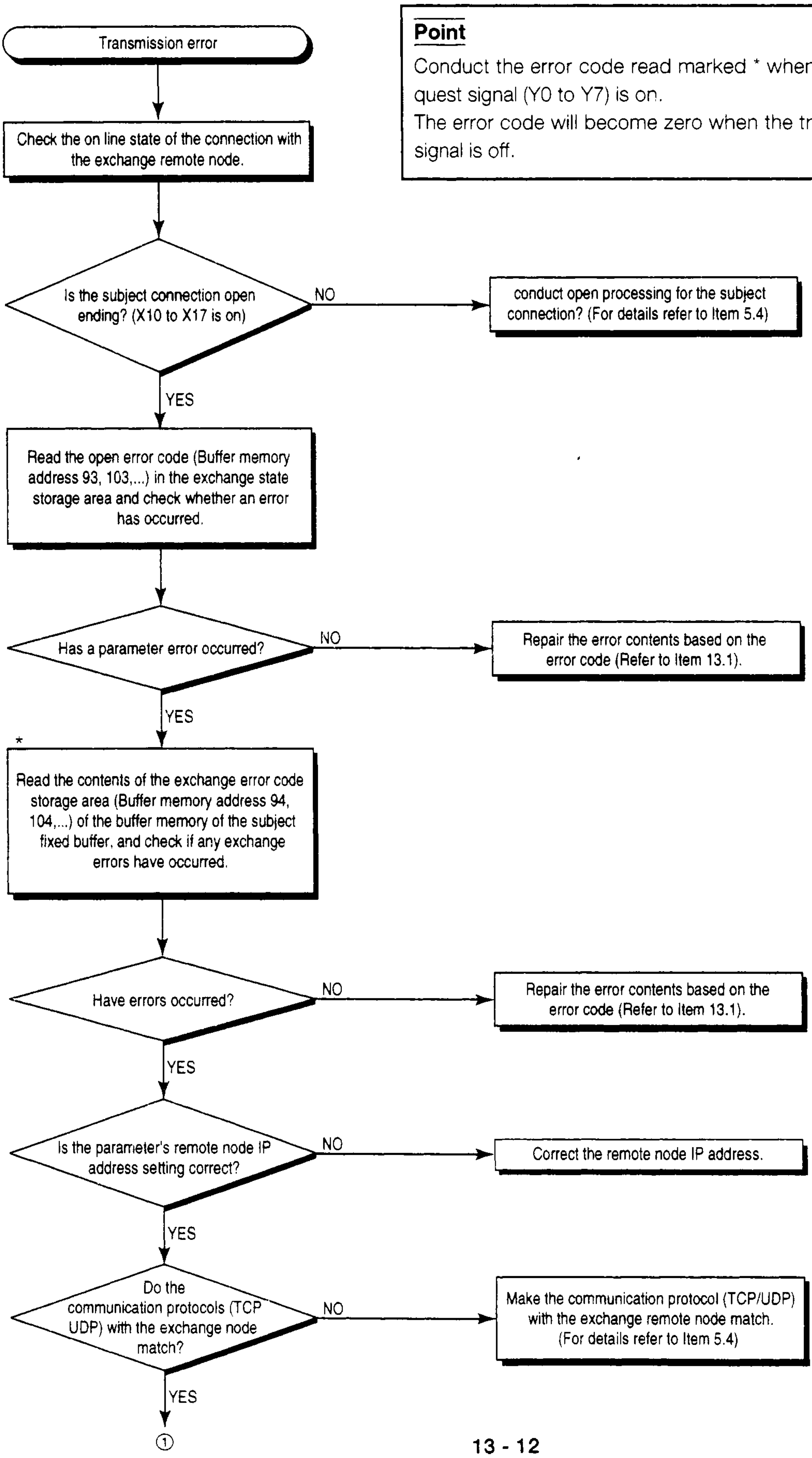
(1) When the E71 is replaced due to an error occurrence, restart the partner nodes listed below and resume data communication.
- All partner nodes that were communicating with the station where the replaced E71 was installed.
- All partner nodes that were communicating with other PLCs via the station where the replaced E71 was installed.

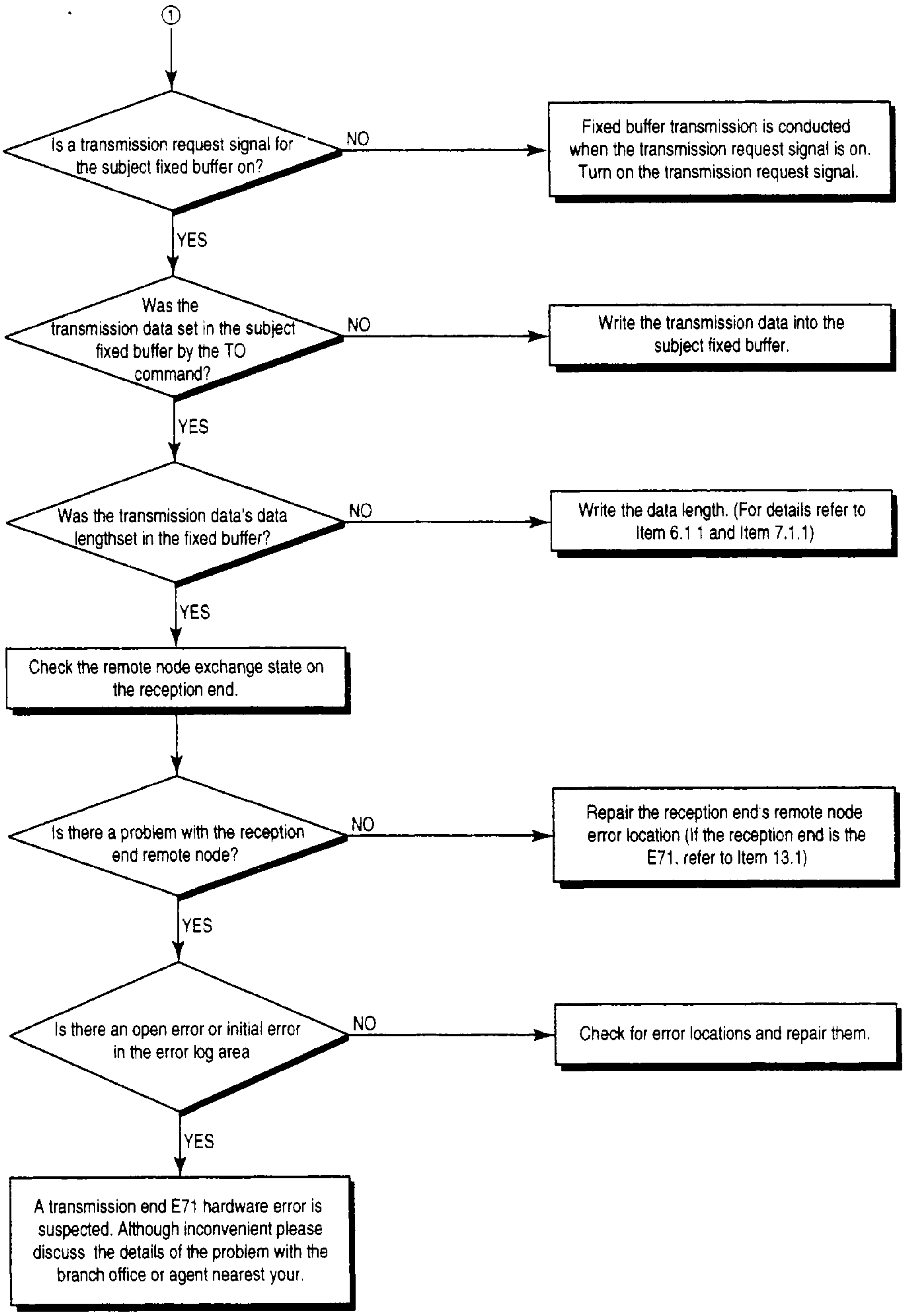
(2) Verify the required devices and connection method by referring to the following when connecting the E71 to Ethernet.
Item 2.3: Verifying the Required Devices
Item 4.7: Verifying the Connection Method

13.2.1 Transmission Error during Fixed Buffer Exchange (Common for Both with Procedure/without Procedure)

Point

Conduct the error code read marked * when the transmission request signal (Y0 to Y7) is on.
The error code will become zero when the transmission reception signal is off.

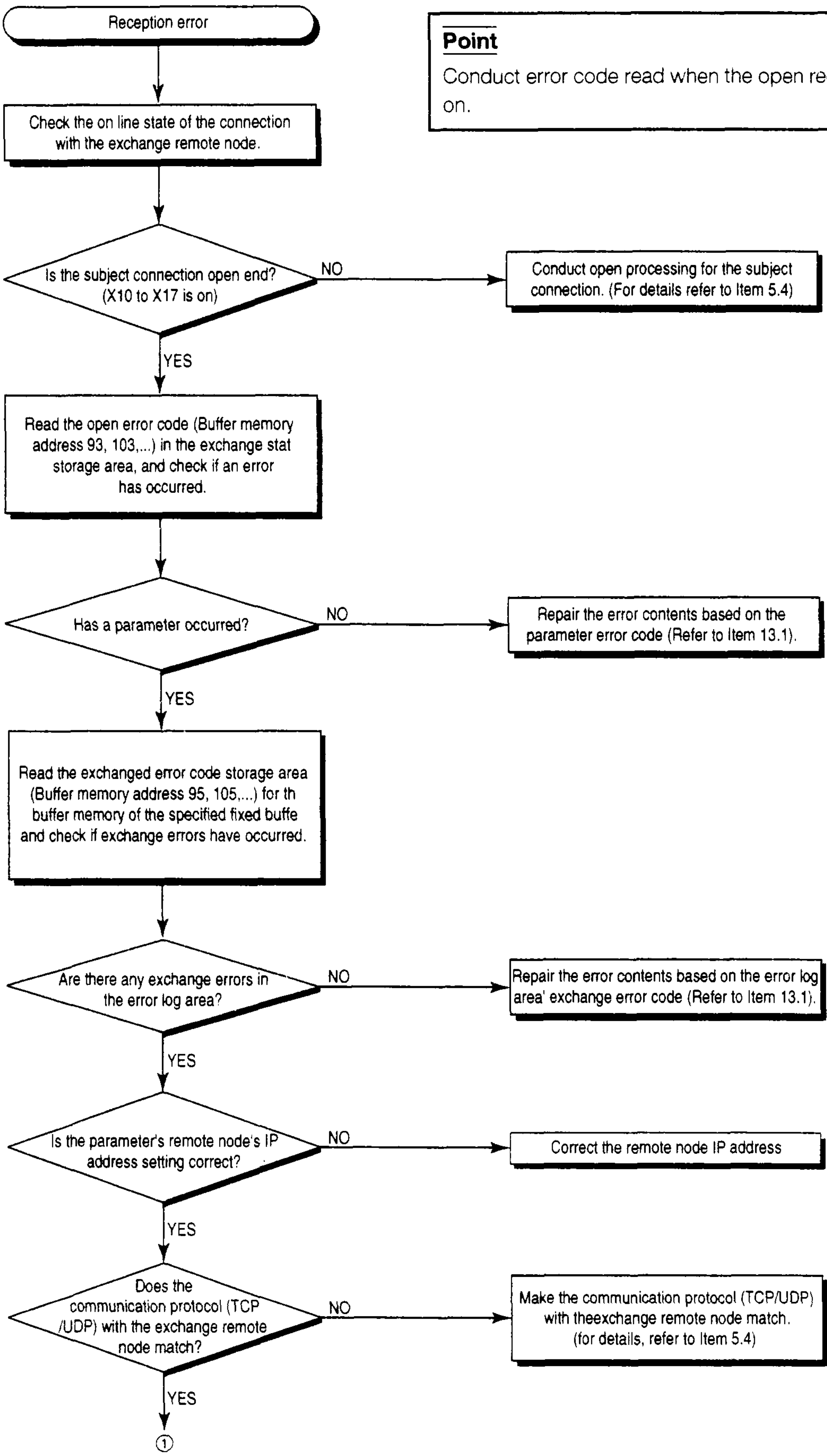


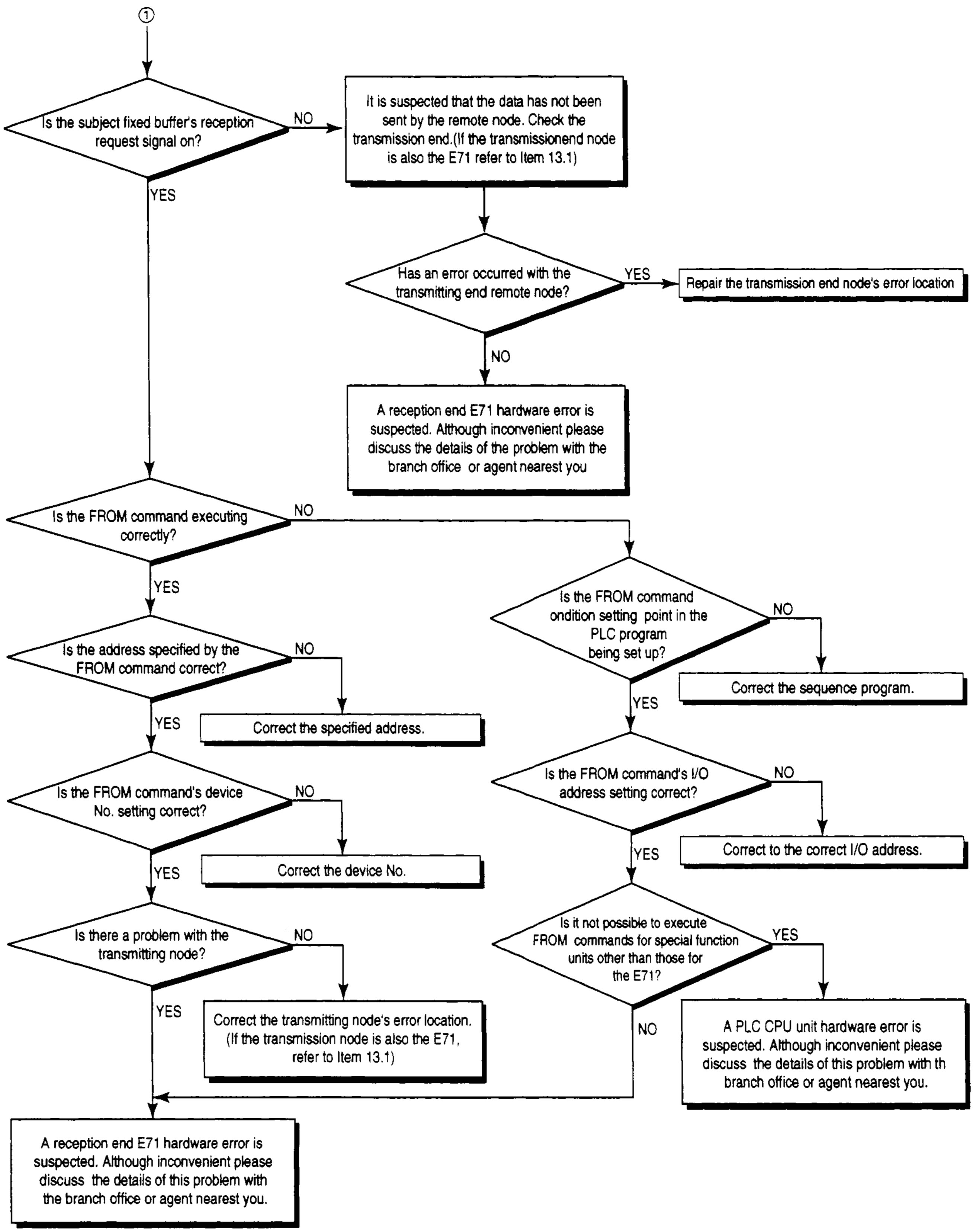

①
Is a transmission request signal for the subject fixed buffer on?
NO
Fixed buffer transmission is conducted when the transmission request signal is on. Turn on the transmission request signal.
YES
Was the transmission data set in the subject fixed buffer by the TO command?
NO
Write the transmission data into the subject fixed buffer.
YES
Was the transmission data's data lengthset in the fixed buffer?
NO
Write the data length. (For details refer to Item 6.1 1 and Item 7.1.1)
YES
Check the remote node exchange state on the reception end.
Is there a problem with the reception end remote node?
NO
Repair the reception end's remote node error location (If the reception end is the E71, refer to Item 13.1)
YES
Is there an open error or initial error in the error log area
NO
Check for error locations and repair them.
YES
A transmission end E71 hardware error is suspected. Although inconvenient please discuss the details of the problem with the branch office or agent nearest your.

13.2.2 Reception Error during Fixed Buffer Exchange (Common for Both with Procedure/without Procedure)

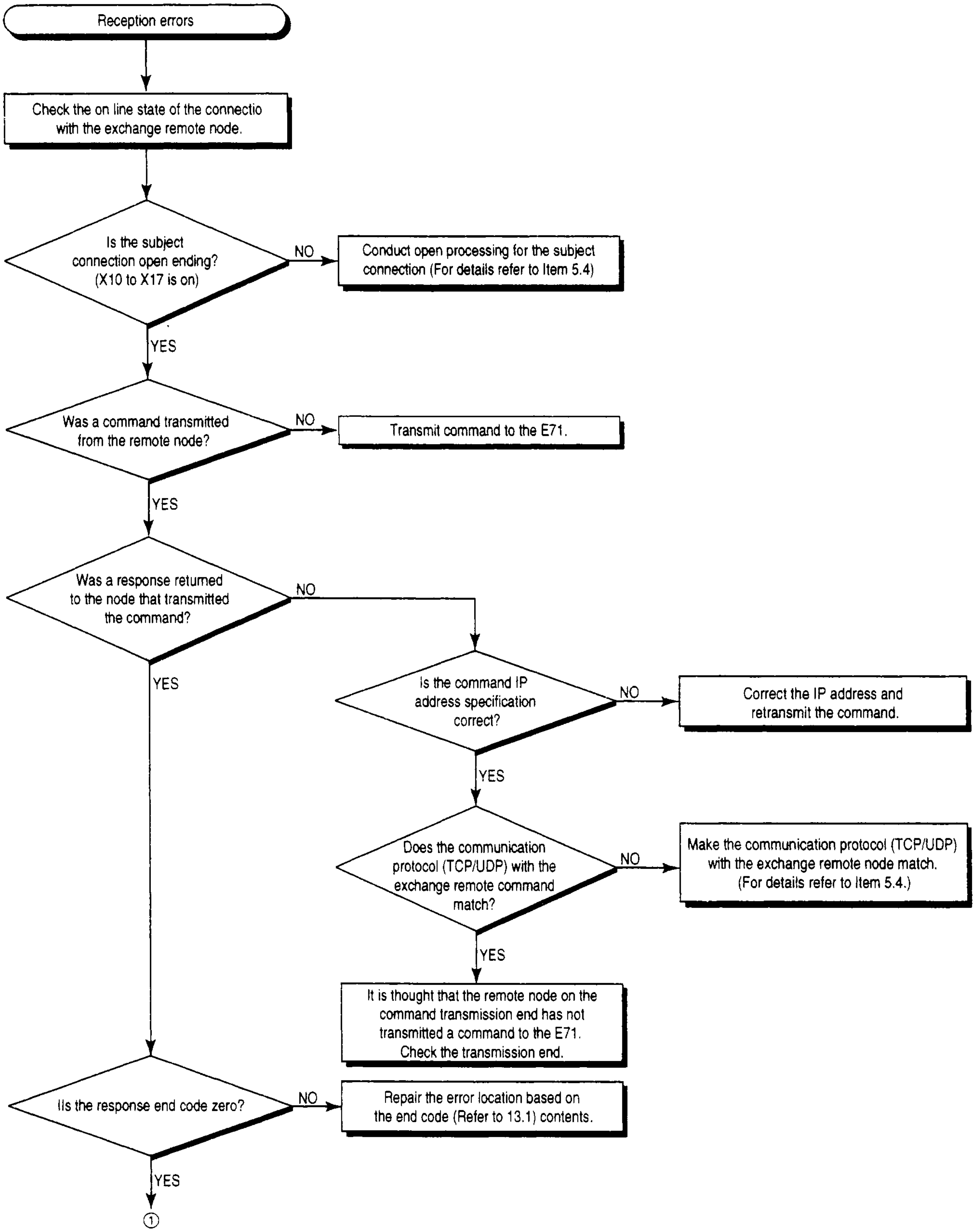
Point

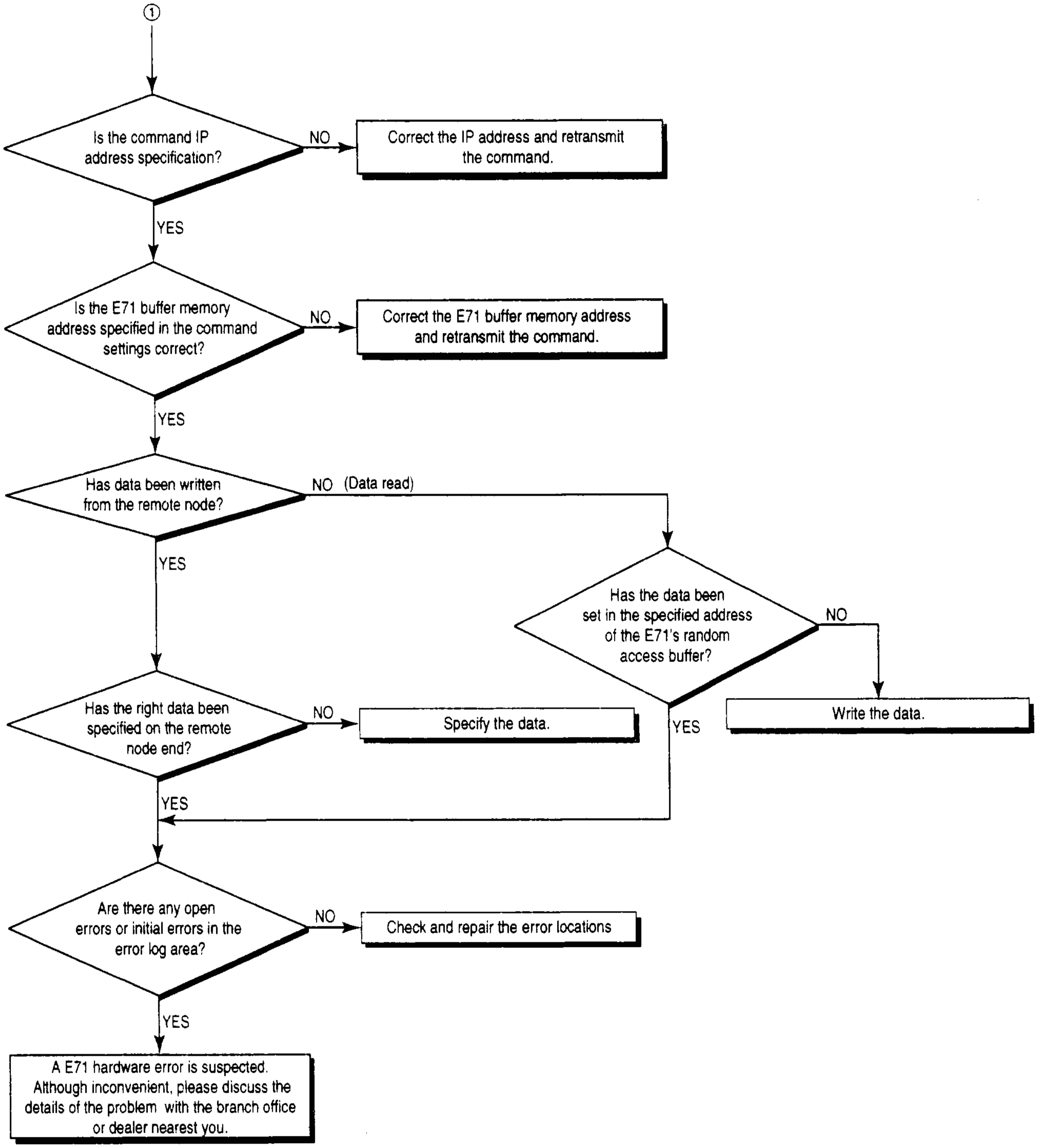
Conduct error code read when the open request signal (Y8 to YF) is on.



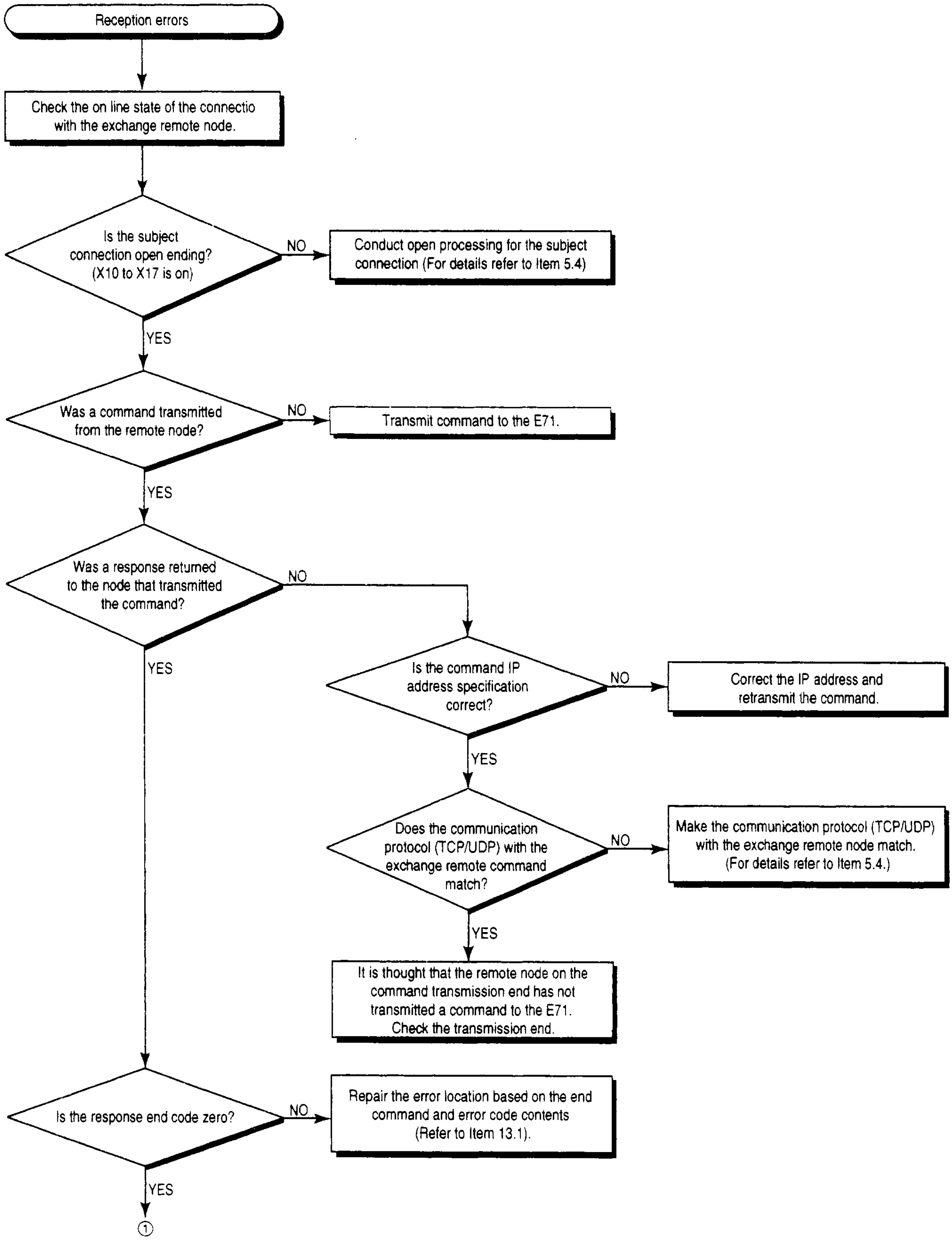

①
Is the subject fixed buffer's reception request signal on?
NO
It is suspected that the data has not been sent by the remote node. Check the transmission end.(If the transmissionend node is also the E71 refer to Item 13.1)
Has an error occurred with the transmitting end remote node?
YES
Repair the transmission end node's error location
NO
A reception end E71 hardware error is suspected. Although inconvenient please discuss the details of the problem with the branch office or agent nearest you
YES
Is the FROM command executing correctly?
NO
Is the FROM command ondition setting point in the PLC program being set up?
NO
Correct the sequence program.
YES
Is the FROM command's I/O address setting correct?
NO
Correct to the correct I/O address.
YES
Is it not possible to execute FROM commands for special function units other than those for the E71?
YES
A PLC CPU unit hardware error is suspected. Although inconvenient please discuss the details of this problem with th branch office or agent nearest you.
NO
YES
Is the address specified by the FROM command correct?
NO
Correct the specified address.
YES
Is the FROM command's device No. setting correct?
NO
Correct the device No.
YES
Is there a problem with the transmitting node?
NO
Correct the transmitting node's error location. (If the transmission node is also the E71, refer to Item 13.1)
YES
A reception end E71 hardware error is suspected. Although inconvenient please discuss the details of this problem with the branch office or agent nearest you.

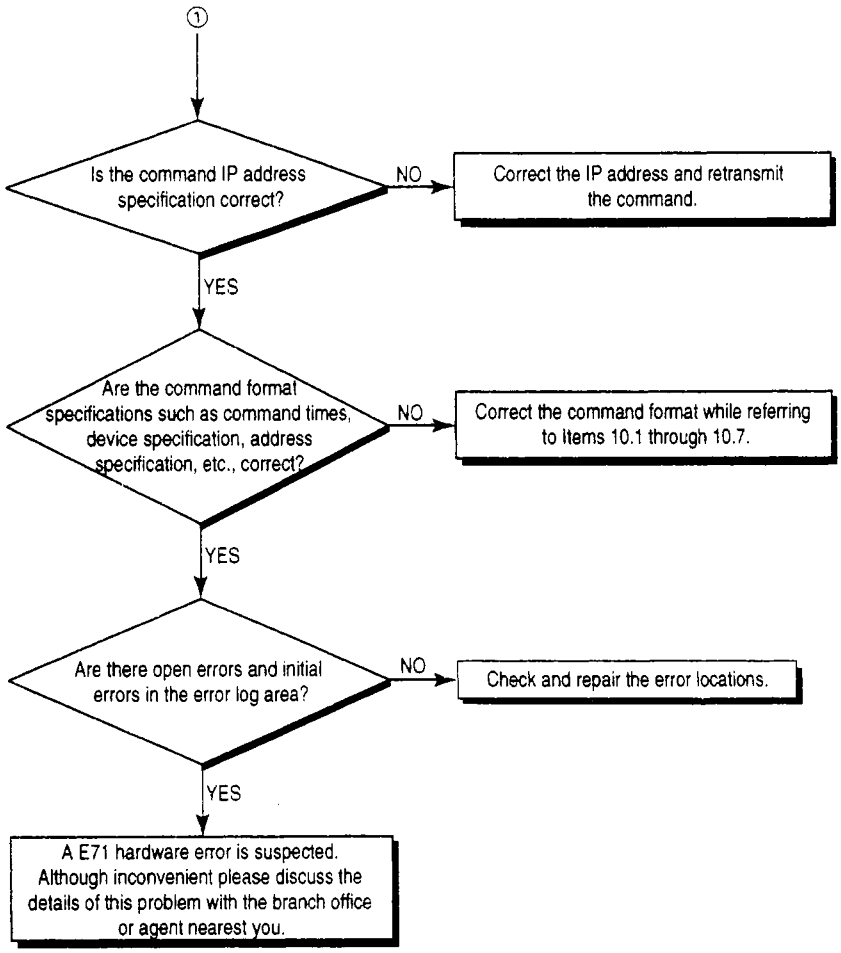
13.2.3 Error during Random Access Buffer Exchange




①
Is the command IP address specification?
NO
Correct the IP address and retransmit the command.
YES
Is the E71 buffer memory address specified in the command settings correct?
NO
Correct the E71 buffer memory address and retransmit the command.
YES
Has data been written from the remote node?
NO (Data read)
YES
Has the data been set in the specified address of the E71's random access buffer?
NO
Write the data.
YES
Has the right data been specified on the remote node end?
NO
Specify the data.
YES
Are there any open errors or initial errors in the error log area?
NO
Check and repair the error locations
YES
A E71 hardware error is suspected. Although inconvenient, please discuss the details of the problem with the branch office or dealer nearest you.

13.2.4 Error When Reading/Writing Data to the PLC CPU




①
Is the command IP address specification correct?
NO
Correct the IP address and retransmit the command.
YES
Are the command format specifications such as command times, device specification, address specification, etc., correct?
NO
Correct the command format while referring to Items 10.1 through 10.7.
YES
Are there open errors and initial errors in the error log area?
NO
Check and repair the error locations.
YES
A E71 hardware error is suspected. Although inconvenient please discuss the details of this problem with the branch office or agent nearest you.

APPENDICES

Appendix 1 Substituting from AJ71E71 (Previous Product)

Data exchange between the PLC CPU and a remote node on the Ethernet conducted by an AJ71E71 (previous product) can be conducted using the E71 (product covered by this manual).

The following explains the hardware and software compatibility when substituting the AJ71E71 with the E71 in a system that conducts data communication using the AJ71E71.

Appendix1.1 Module Compatibility

Following is an explanation of the module compatibility when substituting the E71 for the AJ71E71.

(1) The module (hardware) specifications for the AJ71E71 and the E71 are the same. (They are compatible.)

(2) Data can be exchanged using the same wiring as for the AJ71E71.

(3) The data exchange between the PLC CPU and a remote node on the Ethernet that is conducted by the AJ71E71 can be conducted in the same way by the E71.

Appendix1.2 Program Utilization

The following explains the data communication program compatibility when substituting the E71 for the AJ71E71.

Appendix 1.2.1 Remote Node Side Program Utilization

The following explains the remote node side data communication program compatibility when the A71 is substituted for the AJ71E71.

(1) The program for the following exchange function portion for the AJ71E71 can be utilized to conduct data exchange for the E71. However, the response performance differs somewhat between the AJ71E71 and the E71, so there are times when data exchange cannot be utilized as is. The response timeout time between the E71 and the remote node needs to be adjusted. Be sure to check the operation when utilizing a remote node side program for the AJ71E71.

(2) The functions that can be used for exchange for program utilization are as follows.
The remote node in the table is the node that is conducting data exchange to the AJ71E71.

Utilization program \ Exchange Partner		Remote Node to E71	E71 to Remote Node	E71 to AJ71E71	AJ71E71 to E71
Exchange Functions	Fixed buffer exchange (With procedure)	○	○	○	○
	Random access buffer exchange	○			
	Reading/writing data to the PLC CPU	○			

○ : Exchange is possible by diverting as is the program for the E71 from the remote node

Remarks

When using E71 exchange functions other than those above, create a new exchange program.

(3) The E71 IP address class must be changed to class A through class C. Set the IP address to be set in the E71 and conduct open processing/data exchange.

Appendix 1.2.2 Sequence Program Utilization

The following explains the PLC CPU data exchange program compatibility when substituting the E71 for the AJ71E71.

(1) It is possible to conduct data exchange to a remote node from the E71 by utilizing the fixed buffer exchange (with procedure) function portion of the program for the remote node. However, the response performance between the AJ71E71 and E71 differs somewhat, so there are times the data exchange cannot be utilized as is. It is necessary to adjust the response timeout time between the E71 and the remote node. Be sure to check the operation when utilizing the AJ71E71 sequence program.

(2) The functions for which exchange is possible when utilizing the program are as shown below. The remote nodes shown in the table are remote nodes with which data exchange is being conducted with the AJ71E71.

Utilization program \ Exchange Partner		Remote Node to E71	E71 to Remote Node	E71 to AJ71E71	AJ71E71 to E71
Exchange Functions	Fixed buffer exchange (With procedure)	○	○	○	○
	Random access buffer exchange				
	Reading/writing data to the PLC CPU				

○ : Exchange is possible by diverting as is the program for the AJ71 E71

Remarks

When using E71 exchange functions other than those above, create a new exchange program.

(3) Change the E71 IP address class to class A through class C. Set the IP address to be set in the E71 for the primary remote node and conduct open processing/data exchange.

Appendix 2 Adding the Ethernet Interface Module to the Existing System

The E71 and AJ71E71 can coexist in the same Ethernet. The wiring used for the AJ71E71 can be used as is in the existing system Ethernet that contains the E71.

Appendix 3 Processing Time

Use the following formulas to calculate the minimum transmission delay time for each function.

However, the transmission delay time is sometimes increased by the network load rate (line connections), each node window size, the number of connections used at the same time, and the system configuration. The value found using the following formula is used as the transmission delay time measure when conducting exchange while using one connection.

1 Fixed buffer exchange minimum transmission delay time (When exchanging between E71 and E71)

(a) For TCP/IP

$$47 + (0.025 \times \underbrace{(\text{Command data length})}_{\text{The unit is a byte.}}) + (0.025 \times \underbrace{(\text{Response data length})}_{\text{The unit is a byte.}})$$
$$+ (\text{Transmission scan time}) + (\text{Reception scan time})\ (\text{ms})$$

(b) For UDP/IP

$$47 + (0.023 \times \underbrace{(\text{Command data length})}_{\text{The unit is a byte.}}) + (0.023 \times \underbrace{(\text{Response data length})}_{\text{The unit is a byte.}})$$
$$+ (\text{Transmission scan Time}) + (\text{Reception scan time})\ (\text{ms})$$

Command data length : This is the data length including the subheader, data length, and text data specified in the command application data portion for fixed buffer data transmission. The unit is 1 byte.

	Command Data Length	
	With Procedure	**Without Procedure**
Exchange using binary code	4 + (Data length) × 2	(Number of text bytes)
Exchange using ASCII code	8 + (Data length) × 4	—

Response data length : This is the data length that is fixed in the response application data portion and includes the subheader and end code when receiving data using the fixed buffer. The unit is 1 byte.

	Response Data Length	
	With Procedure	**Without Procedure**
Exchange using binary code	2	—
Exchange using ASCII code	4	—

[Example calculation]

The minimum transmission delay time when transmitting 1017 words of data between the E71 and an E71 when TCP/IP binary code exchange is used for the protocol.

(The transmission scan time is 100ms, and the reception scan time is 80ms.)

$$47 + (0.025 \times (4 + (1017 \times 2))) + (0.025 \times 2) + 100 + 80 \approx 278\ (\text{ms})$$

2 **Random access buffer exchange minimum transmission delay time**

(a) For TCP/IP

$$30+(0.018\times\underbrace{(\text{Command data length})}_{\text{The unit is a byte.}})+(0.007\times\underbrace{(\text{Response data length})}_{\text{The unit is a byte.}})$$
$$+(\text{Remote node ACK processing time})\ (\text{ms})$$

(b) For UDP/IP

$$30+(0.017\times\underbrace{(\text{Command data length})}_{\text{The unit is a byte.}})+(0.006\times\underbrace{(\text{Response data length})}_{\text{The unit is a byte.}})\ (\text{ms})$$

Command data length : This is the data length including the subheader, data length, text data, that is set in the command application data portion when reading and writing is conducted to and from the random access buffer. The unit is 1 byte.

	Command Data Length	
	Read	Write
Exchange using binary code	6	6 + ((Data length) × 2)
Exchange using ASCII code	12	12 + ((Data length) × 4)

Response data length : This is the data length including the subheader and end code, that is set in the response application data portion when reading from and writing to the random access buffer is conducted.

	Response Data Length	
	Read	Write
Exchange using binary code	2 + ((Data length) × 2)	2
Exchange using ASCII code	6 + ((Data length) × 4)	6

Remote node ACK processing : This is the time from when the read/write to or from the random buffer ends until the remote node returns ACK.

[Example calculation 1]

The minimum transmission delay time when 508 words of data are read from the data register (D) using the UDP/IP's ASCII code exchange as the protocol

$30+(0.017\times 12)+(0.006\times(6+(508\times 4)))\approx 43\ (\text{ms})$

[Example calculation 2]

The minimum transmission delay time when 508 words of data are written to the data register (D) using the UDP/IP's ASCII code as the exchange protocol

$30+(0.017\times(12+(508\times 4)))+(0.006\times 6)\approx 65\ (\text{ms})$

3 Minimum transmission delay time when reading/writing data to the PLC CPU

(a) For TCP/IP

$$30 + (0.018 \times \underbrace{(\text{Command data length})}_{\text{The unit is a byte.}}) + (0.007 \times \underbrace{(\text{Response data length})}_{\text{The unit is a byte.}})$$
$$+ (\text{PC CPU processing time}) + (\text{Remote node ACK processing time})\ (\text{ms})$$

(b) For UDP/IP

$$30 + (0.017 \times \underbrace{(\text{Command data length})}_{\text{The unit is a byte.}}) + (0.006 \times \underbrace{(\text{Response data length})}_{\text{The unit is a byte.}})\ (\text{ms})$$
$$+ (\text{PC CPU processing time})$$

Command data length : This is the data length including the subheader, data length, and text data, that is specified in the command application data portion when data is read from or written to the PLC CPU. The unit is 1 byte.
The command data length changes depending on the commands used. Refer to Items 10.1 and 10.2 through 10.7.

Response data length : This is the data length including the subheader and the end command that is set in the response application data portion when data is read from or written to the PLC CPU. The unit is 1 byte.
The command data length varies depending on the commands used. Refer to the Item 10.1 and Items 10.2 through 10.7.

PLC CPU processing time : This is the processing time of the request to read/write data to the PLC CPU. This time is determined by the read/write data, number of points to process, and the PLC CPU scan time.
Refer to (c).

PLC CPU processing time =

$$\underbrace{(\text{Specified number of points}) \div (\text{Number of points processed in one sequence program scan})}_{\text{Rounded off below the decimal point}} \times (\text{Scanner time})$$

Remote node ACK processing : This is the time from the PLC CPU data read/write end time until the remote node returns an ACK.

[Example calculation 1]

The minimum transmission delay time (scan time is 100ms) when reading data register (D) of 100 points of data while using the TCP/IP's ASCII code for the exchange protocol.

Command data length = 24 bytes
Response data length = 404 bytes
PLC CPU process time = (100 ÷ 64) × 100=200 (ms)
Minimum transmission delay time = 30 + (0.018 × 24) + (0.007 × 404) + 200 + (remote node ACK process time)
= 234 + (remote node ACK processing time) (ms)

[Example calculation 2]

The minimum transmission delay time (scanner time is 100ms) when writing 100 points of data to the data register D100 using the TCP/IP's ASCII code as the exchange protocol.

Command data length = 424 bytes
Response data length = 4 bytes
PLC CPU process time = $(100 \div 64) \times 100 = 200$ (ms)
Minimum transmission delay time = $30 + (0.018 \times 424) + (0.007 \times 4) + 200 +$ (remote node's ACK processing time)
= 238 + (remote node's ACK processing time) (ms)

(c) Exchange time with the PLC CPU (scan time extension time)

In response to a request from the E71, the PLC CPU main unit processes the number of process points for 1 scan of the sequence program for each END when running. The intervention time for this scan time and the number of scans required for processing are shown below.

Item					**PLC CPU processing time (scan extension)**		Maximum number of processing points between the E71 and the remote node	**Number of processing points for each sequence program scan**	**Number of scans required for processing**
					AnSCPU A1SJCPU A0J2HCPU AnNCPU	**A2ASCPU AnACPU AnUCPU**			
Device data	Device memory	Batch read	Bit unit		0.76ms	1.38ms	256 points	256 points	1 scan
			Word unit	Bit device	1.13ms	2.42ms	128 words (2048 points)	32 words (512 points)	(Specified number of points/32) scan Decimal point rounded off (Maximum 4 scans)
				Word device	1.13ms	2.42ms	256 points	64 points	Other than device R (Specified number of points/64) scan Decimal point rounded off (Maximum 4 scans) Device R (Specified number of points/64) + 1 scan Decimal point rounded off (Maximum 5 scans)
		Batch write	Bit unit		1.13ms	1.06ms	256 points	256 points	2 scans (1 scan when set to possible during RUN.)
			Word unit	Bit device	1.13ms	2.60ms	40 words (640 points)	10 words (160 points)	(Specified number of points/10) + 1 scan Decimal point rounded off "0" when set to possible during RUN (Maximum 5 scans)
				Word device	1.13ms	2.60ms	256 points	64 points	Other than device R (Specified number of points/64) + 1 scan Decimal point rounded off "0" when set to possible during RUN (Maximum 5 scans) Device R (Specified number of points/64) + 1 scan Decimal point rounded off (Maximum 5 scans)

Item				PLC CPU processing time (scan extension) AnSCPU A1SJCPU A0J2HCPU AnNCPU	A2ASCPU AnACPU AnUCPU	Maximum number of processing points between the E71 and the remote node	Number of processing points for each sequence program scan	Number of scans required for processing
Device data	Device memory	Test (random write)	Bit unit	1.13ms	1.06ms	80 points	20 points	(Specified number of points/20) + 1 scan Decimal point rounded off '0' when set to possible during RUN. (Maximum 5 scans)
			Word unit – Bit device	1.13ms	1.06ms	40 words (640 points)	10 words (160 points)	(Specified number of points/10) + 1 scan Decimal point rounded off '0' when set to possible during RUN (Maximum 5 scans)
			Word unit – Word device	1.13ms	1.06ms	40 points	10 points	Other than device R (Specified number of points/10) + 1 scan Decimal point rounded off '0' when set to possible during RUN. (Maximum 5 scans) Device R (Specified number of points/10) + 1 scan Decimal point rounded off (Maximum 5 scans)
		Monitor data register	Bit unit	—	—	—	—	———
			Word unit			—	—	1 scan for device R only
		Monitor	Bit unit	2.02ms	1.46ms	40 points	40 points	1 scan
			Word unit – Bit device	2.08ms	1.47ms	320 points (20 words)	320 points (20 words)	1 scan
			Word unit – Word device	2.08ms	1.47ms	20 points	20 points	
	Extension file register	Batch read		1.27ms	2.42ms	256 points	64 points	(Specified number of points/64) + 1 scan Decimal point rounded off (Maximum 5 scans)
		Batch write		1.27ms	2.60ms	256 points	64 points	
		Direct read		—	2.30ms	256 points	64 points	
		Direct write		—	2.57ms	256 points	64 points	
		Test (random write)		1.31ms	0.97ms	40 points	10 points	
		Monitor data register		—		—	—	———
		Monitor		1.75ms	1.42ms	20 points	20 points	1 scan
Special function module buffer memory	Batch read			FROM instruction process-ing time + 1.13 ms	FROM instruction process-ing time + 0.75 ms	256 bytes	128 bytes	(Specified number of points/128) scan Decimal point rounded off (Maximum 2 scans)
	Batch write							(Specified number of points/128) + 1 scan Decimal point rounded off '0' when set to possible during RUN. (Maximum 3 scans)

Item				PLC CPU processing time (scan extension) AnSCPU A1SJCPU A0J2HCPU AnNCPU	PLC CPU processing time (scan extension) A2ASCPU AnACPU AnUCPU	Maximum number of processing points between the E71 and the remote node	Number of processing points for each sequence program scan	Number of scans required for processing
Program	Sequence program	Batch read	Main	1.20ms	0.70ms	256 steps	64 steps	(Specified number of steps/64) scan Decimal point rounded off (Maximum 4 scans)
			Sub	1.20ms	0.70ms			
		Batch write	Main	0.67ms	0.49ms	256 steps	64 steps	(Specified number of steps/64) + 1 scan Decimal point rounded off "0" when set to possible during RUN (Maximum 5 scans)
			Sub	0.67ms	0.49ms			
	Micro-computer program	Batch read	Main	1.35ms	—	256 bytes	128 bytes	(Specified number of bytes/128) + 1 scan Decimal point rounded off (Maximum 3 scans)
			Sub	1.35ms				
		Batch write	Main	1.35ms				
			Sub	1.53ms				
	Comment	Batch read		1.35ms	2.42ms	256 bytes	128 bytes	(Specified number of bytes/128) + 1 scan Decimal point rounded off (Maximum 3 scans)
		Batch write		1.53ms	2.60ms			
	Extension comment	Batch read		—	2.31ms	256 bytes	128 bytes	(Specified number of bytes/128) + 1 scan Decimal point rounded off (Maximum 3 scans)
		Batch write		—	2.59ms			
	Parameter	Batch read		0.68ms	2.42ms	256 bytes	128 bytes	(Specified number of bytes/128) + 1 scan Decimal point rounded off (Maximum 3 scans)
		Batch write		—	—	—	—	——
		Analysis request						
PLC CPU		Remote RUN		—	—	—	—	——
		Remote STOP						
		PLC model name read		—	—	—	—	1 scan

Point

(1) Because the PLC CPU can only process one of the above items during END processing, for the corresponding PLC CPU to access the A6GPP, E71, etc. at the same time it must wait until the other processing is finished. This further increases the number of scans required for processing.

(2) The scan time is extended by approximately 0.2 msec (A2AS, AnA, and AnUCPU are 0.1 msec) even when the E71 is not installed and link is not conducted.

Appendix 4 ASCII Code Table

LSD \ MSD		0	1	2	3	4	5	6	7
		000	001	010	011	100	101	110	111
0	0000	NUL	DLE	SP	0	@	P	'	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	!!	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	/	7	G	W	g	w
8	1000	BS	CAN	(	8	H	X	h	x
9	1001	HT	EM	)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[	k	{
C	1100	FF	FS	,	<	L	⁄	l	|
D	1101	CR	GS	—	=	M	]	m	}
E	1110	SO	RS	.	>	N	↑	n	~
F	1111	SI	VS	/	?	O	←	o	DEL

Appendix 5 Reference Documents

For details regarding TCP/IP refer to the DDN Protocol Handbook (3 volumes).

Publisher

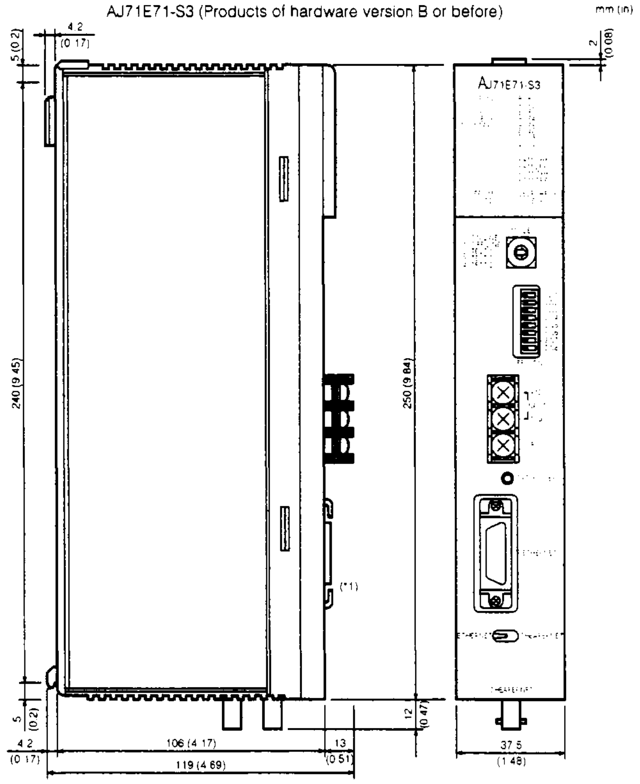
DDN Network Information Center
SRI International
333 Ravenswood Avenue, EJ291
Menlo Park, California 94025

RFC No.

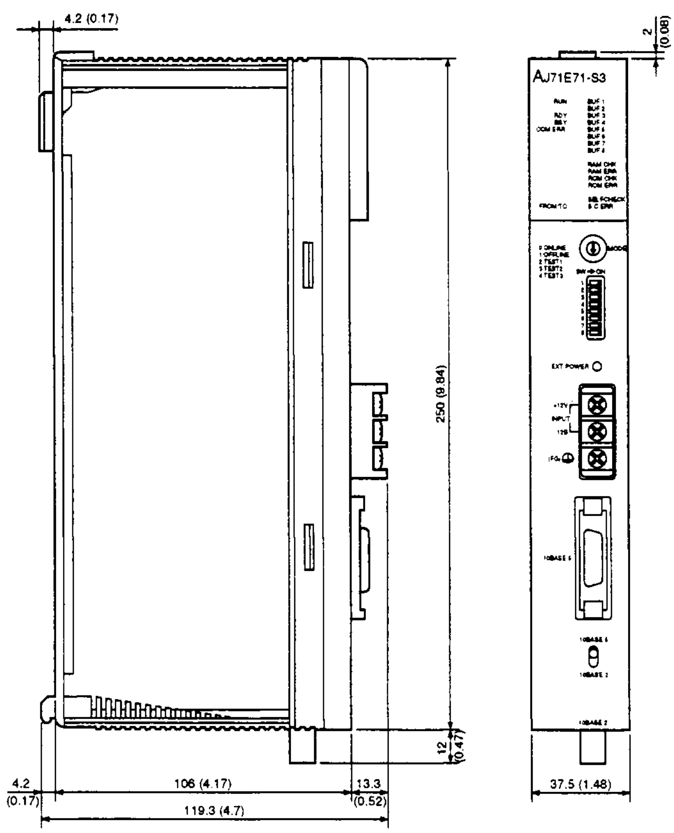
TCP RFC793
UDP RFC768
IP RFC791
ICMP RFC792
ARP RFC826

Appendix 6 Diagram of External Dimensions

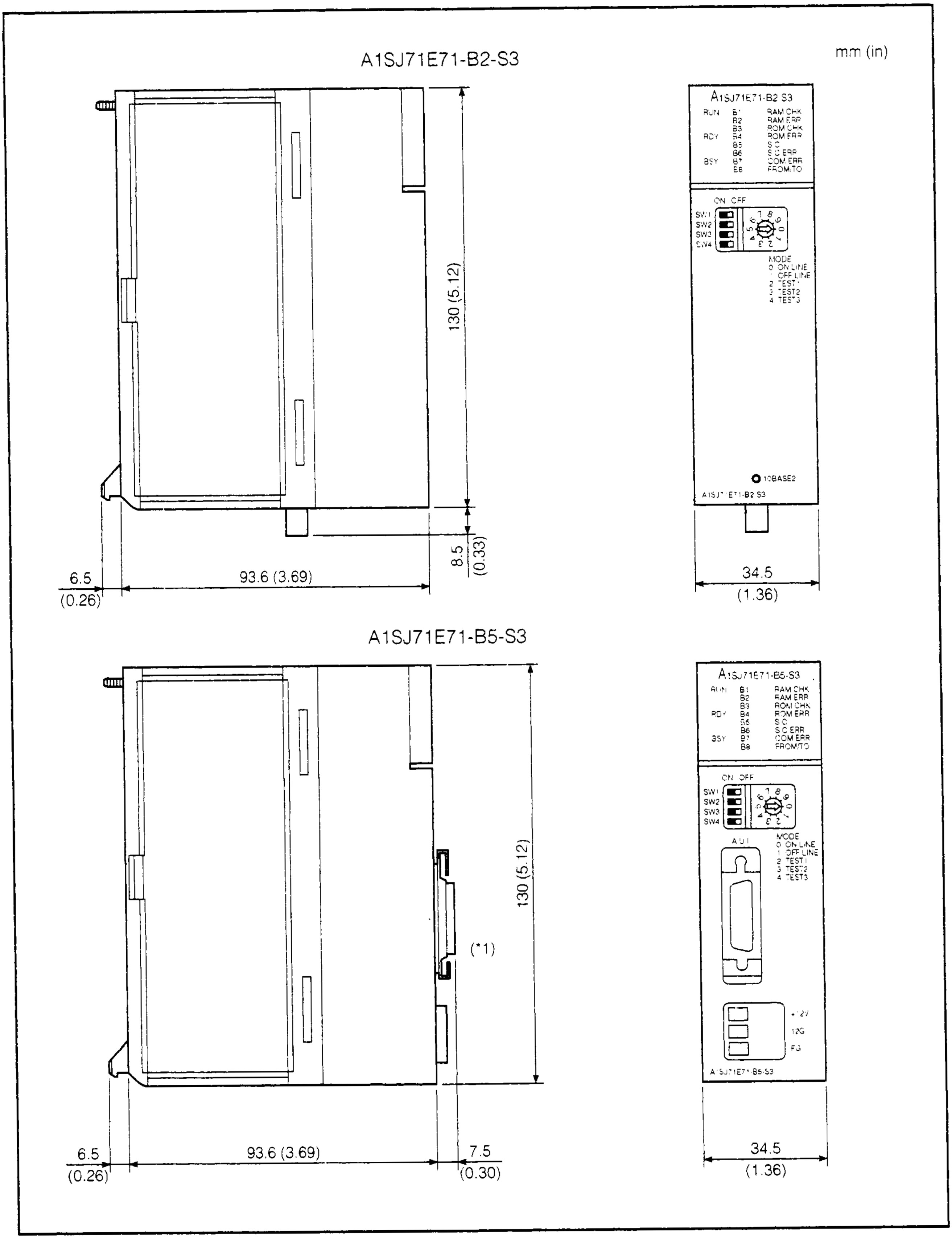
AJ71E71-S3 (Products of hardware version B or before) mm (in)



AJ71E71-S3 (Products of hardware version C or later)



*1 The connector area radius bending ration during cable connection (R1: measured value) is (cable radius x 4) or larger.



*1 The connector area radius bending ration during cable connection (R1: measured value) is (cable radius x 4) or larger.

Appendix 7 Sample Program

This shows the sample program between the PLC CPU of the station installed in the E71 and the remote node in order to conduct a connection test between the E71 (AJ71E71-S3) and a remote node (IBM-PC/AT) connected to the same Ethernet.

The programs only conduct the minimum processing required to conduct the exchange test. Change the IP address and the port No. to match those of the system. In addition, make separate additions when error processing, etc., is included.

- PLC CPU side : Make additions as explained in Chapters 5 and 13 of this manual.
- IBM-PC/AT side : Make additions to match the system specifications.

Appendix 7.1 Program for Reading/Writing Data in the PLC CPU

The sample program or execution environment and data exchagne contents are shown below.

1 Sample program execution environment

(a) PLC CPU side (*1)

① PLC CPU model name of the station installed in the E71 : A3UCPU
② E71 I/O signal : X/Y000 to X/Y01F
③ Ethernet address : Setting not required because this is an ARP function
④ E71 IP address : C0.00.01.FD$_H$ (192.00.01.253)
⑤ E71 port No. : 2000$_H$ (8192)

(b) Remote node (IBM-PC/AT side)

① Operation environment : Windows 95
② Ethernet interface board name : Board that supports WINSOCK
③ Library : WSOCK32.LIB
④ Software development environment : Uses Microsoft Visual C++ (Ver. 4.0)
⑤ Ethernet address : Setting not required because this is an ARP function
⑥ IP address : Reception when opening Active
⑦ Port No. : Reception when opening Active

(c) Communication protocol : TCP/IP

2 Sampling program overview

(a) PLC CPU side PLC program
Only conducts initial processing and open processing.

(b) Remote node (IBM-PC/AT) side program
The above library is used to conduct exchange for reading/writing data in the following PLC CPU.
- Word unit write (D0 to D4 5 points) .... : Refer to Item 10.2.5
- Word unit read (D0 to D4 5 points) .... : Refer to Item 10.2.3

(c) When exchanging ASCII code data is exchanged.

(*1) The E71 switch settings are as follows.

Setting switch			Set value	Setting description
Operation mode setting switch			0	Online
Exchange condition setting switches	SW1	Line processing selection during TCP timeout error	OFF	Closes
	SW2	Data code setting	ON	ASCII code
	SW7	CPU exchange timing setting	ON	Write enable
	SW8	Initial timing setting	OFF	Quick start

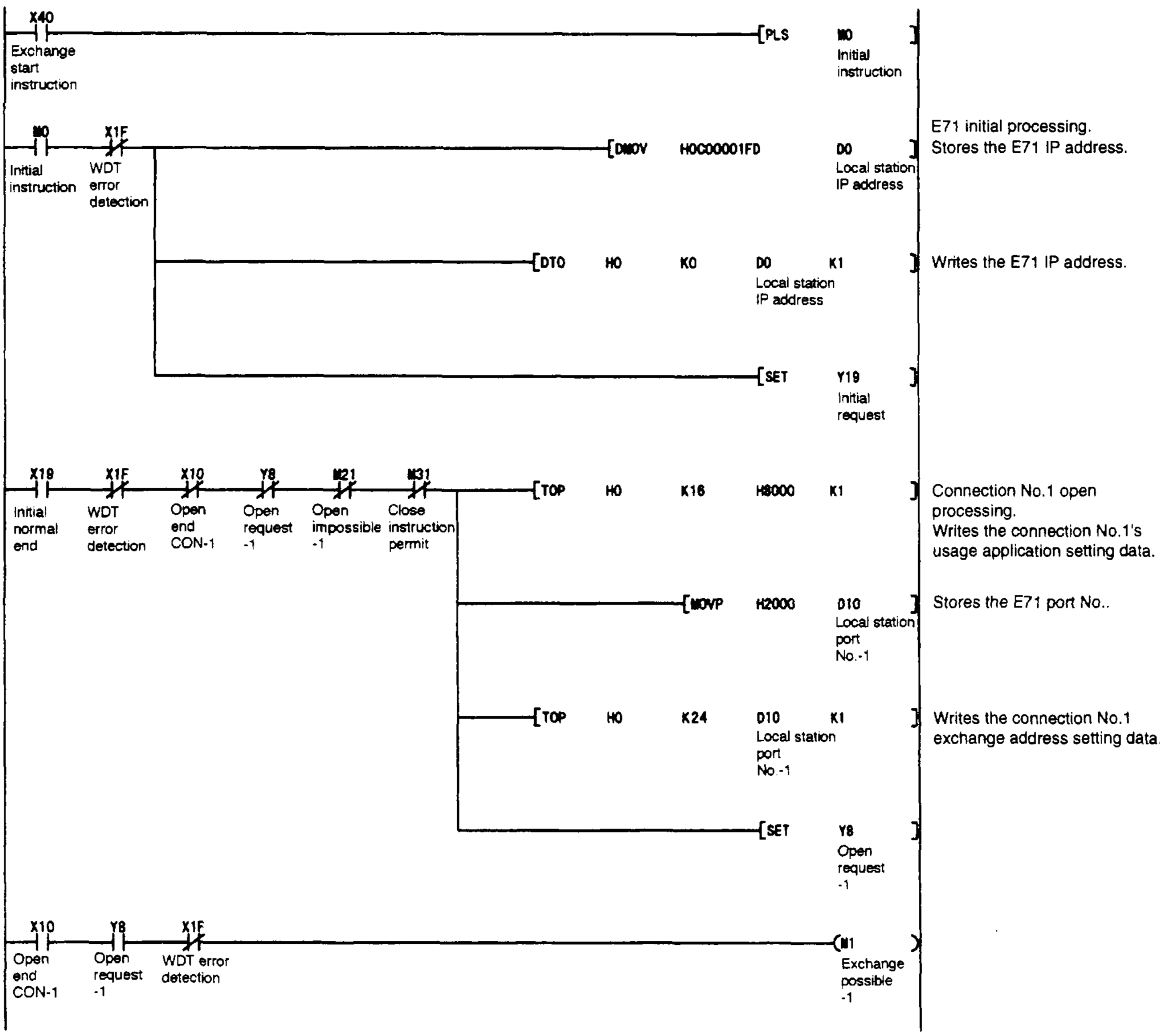
(Set all switches SW3 to SW6 (use not possible) to off.)

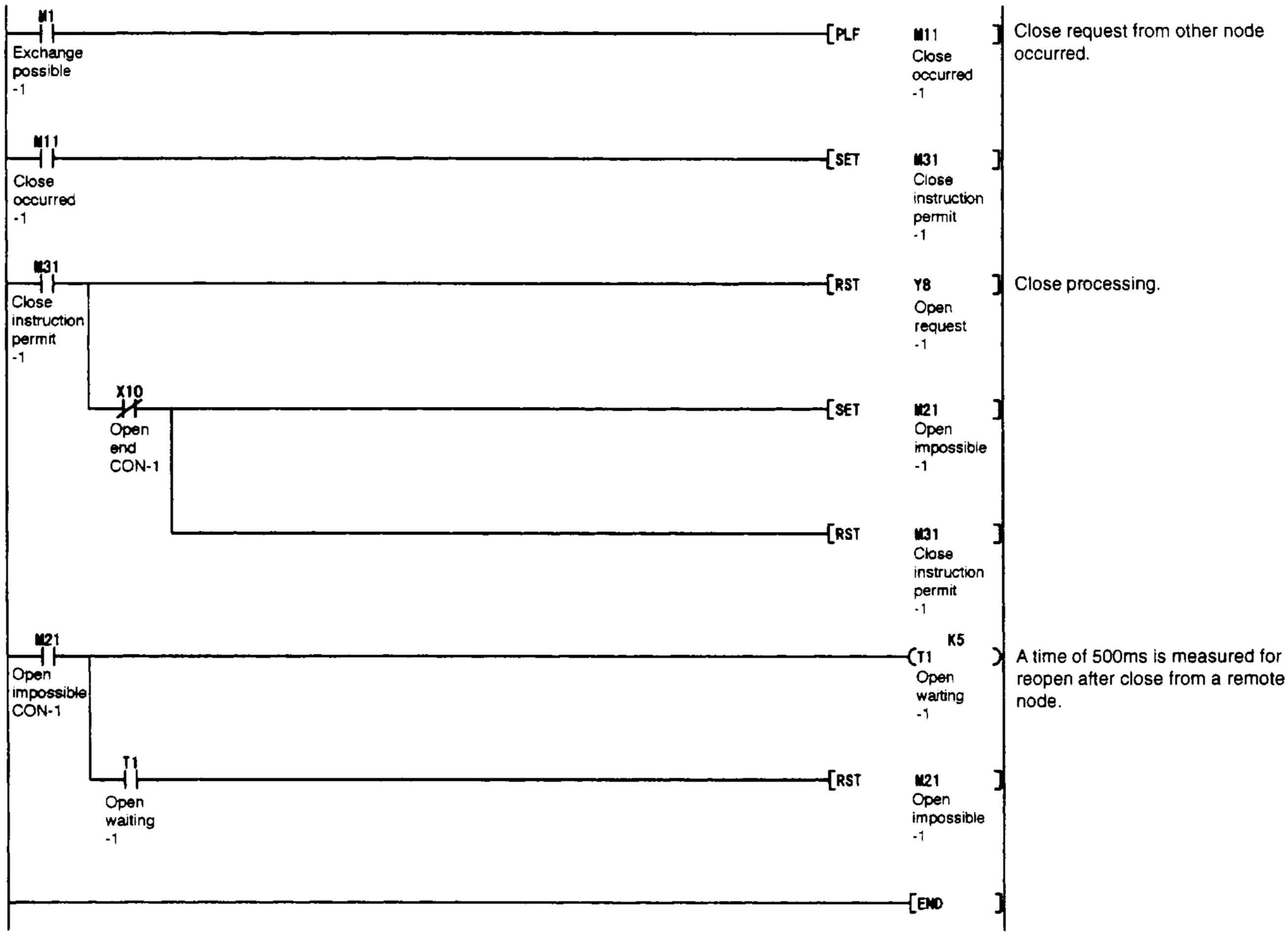
Windows and Microsoft are registered trademarks of the Microsoft Co., LTD.

Other company names and product names are trademarks and registered trademarks in which the corresponding companies have rights.

3 Sequence program

Following is an example of an exchange partner E71 installed station A3UCPU sequence program.




M1
Exchange possible -1
PLF
M11
Close occurred -1
Close request from other node occurred.
M11
Close occurred -1
SET
M31
Close instruction permit -1
M31
Close instruction permit -1
RST
Y8
Open request -1
Close processing.
X10
Open end CON-1
SET
M21
Open impossible -1
RST
M31
Close instruction permit -1
M21
Open impossible CON-1
T1
K5
Open waiting -1
A time of 500ms is measured for reopen after close from a remote node.
T1
Open waiting -1
RST
M21
Open impossible -1
END

4 **Remote node (IBM-PC/AT) side program**

Following is an example remote node program for accessing an A3UCPU while the station installed in the E71.

Executing this program displays in order the following exchange message contents.

① Word unit batch write command message

② Word unit batch write response message

③ Word unit batch read command message

④ Word unit batch read response message

Remarks

(1) Following is a summary of the compiling procedure for the program created using Microsoft Visual C++ (Ver. 4.0).

① Boot up Visual C++.

② Conduct program creation preparation.
Select file to new file and create a console application from the project work space screen.

③ Open the AJSAMP.C file and create a program. (Refer to the example program on the next page.)

④ Execute compiling for the program created from the build menu compile screen.

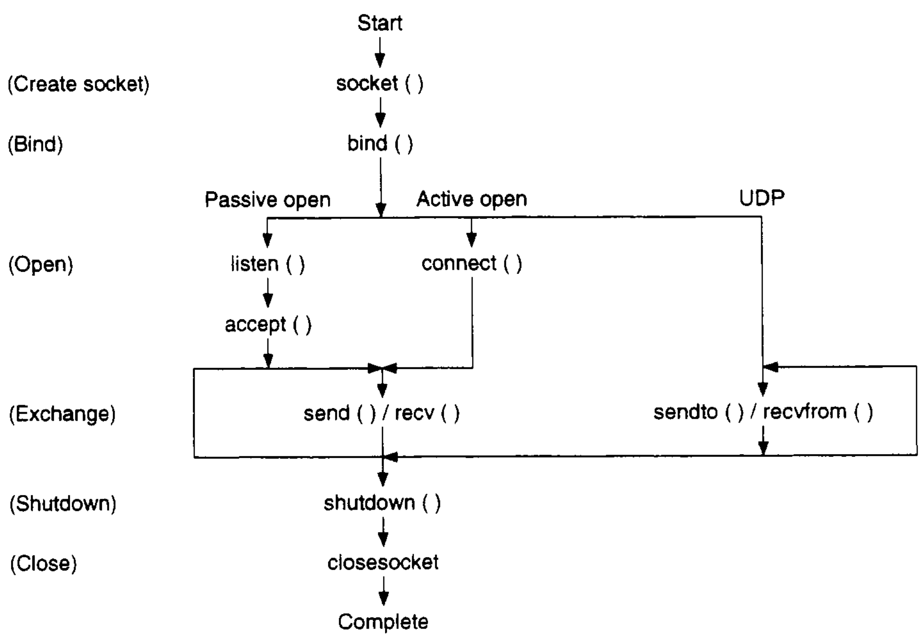
⑤ Link the WSOCK.LIB from the build menu setting screen.

⑥ Create an execute file (AJSAMP.EXE) at the build menu build screen.

⑦ End Visual C++.

⑧ Execute AJSAMP.EXE.

(2) General procedure for socket routine call



```
/********************************************************************************/
/**                              AJSAMP. C                                     **/
/**        AJ71E71-S3 sample program                                           **/
/**                                                                            **/
/**        This program is a sample program for conducting connection          **/
/**        tests between the AJ71E71-S3 and a personal computer.               **/
/**        This program accesses the data register (D) of the PLC CPU          **/
/**        installed in the AJ71E71-S3.                                        **/
/**                                                                            **/
/**                                                                            **/
/**                                                                            **/
/**        Copyright (C) 1996 Mitsubishi Electric Corporation                  **/
/**        All Rights Reserved                                                 **/
/********************************************************************************/

#include    <stdio.h>
#include    <winsock.h>

#define     FLAG_OFF                     0 //End flag off
#define     FLAG_ON                      1 //End flag on
#define     SOCK_OK                      0 //Normal end
#define     SOCK_NG                     -1 //Error end
#define     BUF_SIZE                    80 //Buffer size

#define     ERROR_INITIAL                0 //Initial error
#define     ERROR_SOCKET                 1 //Socket creation error
#define     ERROR_BIND                   2 //Bind error
#define     ERROR_CONNECT                3 //Connect error
#define     ERROR_SEND                   4 //Transmission error
#define     ERROR_RECEIVE                5 //Reception error
#define     ERROR_SHUTDOWN               6 //Shutdown error
#define     ERROR_CLOSE                  7 //Line close error

typedef     struct sck_inf{
            struct in_addr  my_addr;
            unsigned short  my_port;
            struct in_adder aj_addr;
            unsigned short  aj_port;
            };

int nErrorStatus;                           //The error information storage variable
int Dmykeyin;                               //Dummy key input
int Closeflag;                              //Connection end flag
int socketno;
```

```
int main()
{
        WORD wVersionRequested=MAKEWORD(1,1);      //Windows Ver 1.1 Request
        WSADATA wsaData;
        int length;                                //Exchange data length
        unsigned char s_buf[BUF_SIZE];             //Transmission buffer
        unsigned char r_buf[BUF_SIZE];             //Reception buffer
        struct sck_inf sc;
        struct sockaddr_in hostdata;               //Personal computer side data
        struct sockaddr_in aj71e71;                //AJ71E71 side data
        void Sockerror(int);                       //Error processing function

        sc.my_addr.s_addr=htonl(INADDR_ANY);       //Personal computer side IP address
        sc.my_port=htons(0);                       //Personal computer side port No.
        sc.aj_addr.s_addr=inet_addr("192.0.1.253"); //E71 side IP address (C00001FDh)
        sc.aj_port=htons(0x2000);                  //E71 side port No.

        Closeflag=FLAG_OFF;                        //Connection end flag off

        //-----Winsock initial processing-----
        nErrorStatus=WSAStartup(wVersionRequested,&wsaData);

        in(nErrorStatus!=SOCK_OK){
                Sockerror(ERROR_INITIAL);          //Error processing
                return(SOCK_NG);
                }

        printf("Winsock Version is %1d.%1d\n",HIBYTE(wsaData.wVersion),
LOBYTE(wsaData.wVersion));
        printf("AJ_test Start\n");


        socketno=socket(AF_INET,SOCK_STREAM,0);    //TCP/IP socket creation

        if(socketno==INVALID_SOCKET){
                Sockerror(ERROR_SOCKET);           //Error processing
                return(SOCK_NG);
                }

        hostdata.sin_family=AF_INET;
        hostdata.sin_port=sc.my_port;
        hostdata.sin_addr.s_addr=sc.my_addr.s_addr;
                                                   //Bind
        if(bind(socketno,(LPSOCKADDR)&hostdata,sizeof(hostdata))!=SOCK_OK){
                Sockerror(ERROR_BIND);             //Error processing
                return(SOCK_NG);
                }

        aj71e71.sin_family=AF_INET;
        aj71e71.sin_port=sc.aj_port;
        aj71e71.sin_addr.s_addr=sc.aj_addr.s_addr;
```

```
            //-----Connect (Active open) request-----
if(connect(socketno,(LPSOCKADDR)&aj71e71,sizeof(aj71e71))!=SOCK_OK){
            Sockerror(ERROR_CONNECT);          //Error processing
            return(SOCK_NG);
            }

 Closeflag=FLAG_ON;                            //Connection end flag on

            //-----D0 to D4 batch write request-----
strcpy(s_buf,"03FF000A4420000000000500112233445566778899AA");

length=strlen(s_buf);

 if(send(socketno,s_buf,length,0)==SOCKET_ERROR){     //Data transmission
            Sockerror(ERROR_SEND);                    //Error processing
            return(SOCK_NG);
            }
printf("\n Transmission data \n%s\n",s_buf);

 length=recv(socketno,r_buf,BUF_SIZE,0);       //Response data reception

if(length==SOCKET_ERROR){
            Sockerror(ERROR_RECEIVE);          //Error processing
            return(SOCK_NG);
            }
r_buf[length]='\0';                  //Set NULL for the tail of the reception
data
printf("\n Reception data \n%s\n",r_buf);


            //-----D0 to D4 batch read request-----
strcpy(s_buf,"01FF000A4420000000000500");

length=strlen(s_buf);

 if(send(socketno,s_buf,length,0)==SOCKET_ERROR){     //Data transmission
            Sockerror(ERROR_SEND);                    //Error processing
            return(SOCK_NG);
            }
printf("\n Transmission data \n%s\n",s_buf);

 length=recv(socketno,r_buf,BUF_SIZE,0);       //Response data reception

if(length==SOCKET_ERROR){
            Sockerror(ERROR_RECEIVE);          //Error processing
            return(SOCK_NG);
            }
r_buf[length]='\0';                  //Set NULL for the tail of the reception
data
```

```
        printf("\n Reception data \n%s\n",r_buf);

        if(shutdown(socketno,2)!=SOCK_OK){          //Transmission/reception prohibit processing
                Sockerror(ERROR_SHUTDOWN);          //Error processing
                return(SOCK_NG);
                }

                //-----Close processing-----
        if(closesocket(socketno)!=SOCK_OK)
                Sockerror(ERROR_CLOSE);             //Error processing
                return(SOCK_NG);
                }

        Closeflag=FLAG_OFF;                         //Connection end flag off
        WSACleanup();                               //Winsock.DLL Release

        printf("\nAJ_test End.\n\n Ended normally \n");
        printf("Program ends. Push any key.  \n");
        Dmykeyin=getchar();                         //Key input wait
        return(SOCK_OK);
}

void Sockerror(int error_kind)                      //Error processing function
{
        if(error_kind==ERROR_INITIAL){
                printf("Initial processing is abnormal.");
                }
        else{
                nErrorStatus=WSAGetLastError();
                switch(error_kind){
                case ERROR_SOCKET:
                     printf("Socket could not be created.");
                     break;
                case ERROR_BIND:
                     printf("Bind could not be done.");
                     break;
                case ERROR_CONNECT:
                     printf("Connection could not be established.");
                     break;
                case ERROR_SEND:
                     printf("Transmission could not be conducted.");
                     break;
                case ERROR_RECEIVE:
                     printf("Reception could not be conducted.");
                     break;
                case ERROR_SHUTDOWN:
                     printf("Shutdown could not be conducted.");
                     break;
                case ERROR_CLOSE:
                     printf("Normal close could not be conducted.");
                     break;
```

```
            }
        }
        printf("The error code is %d. \n",nErrorStatus);

        if(Closeflag==FLAG_ON){
            nErrorStatus=shutdown(socketno,2);       //Shutdown processing
            nErrorStatus=closesocket(socketno);  //Close processing
            Closeflag=FLAG_OFF;                  //Connection end flag off
            }

        printf("Ends the program. Push any key. \n");
        Dmykeyin=getchar();                      //Key input wait
        WSACleanup();                            //Winsock.DLL Release
        return;
}
```

Appendix 7.2 Sequence Programs for All Functions

An example of a common sequence program for conducting exchange (with procedure) using a fixed buffer memory, exchange using a random access buffer memory, and data read/write in the PLC CPU is shown below.

1 Sample program execution environment

The execution environment for the PLC CPU side setting value and switch setting, etc., is the same as the execution environment shown in Appendix 7.1 Item 1 (a).

In addition, the port No. 2001H(8193) is used.

Freely set the IP Address on the remote node side. Use the same No. for the port No. as for the PLC CPU side.

2 Sample program overview

① Conducts initial processing.

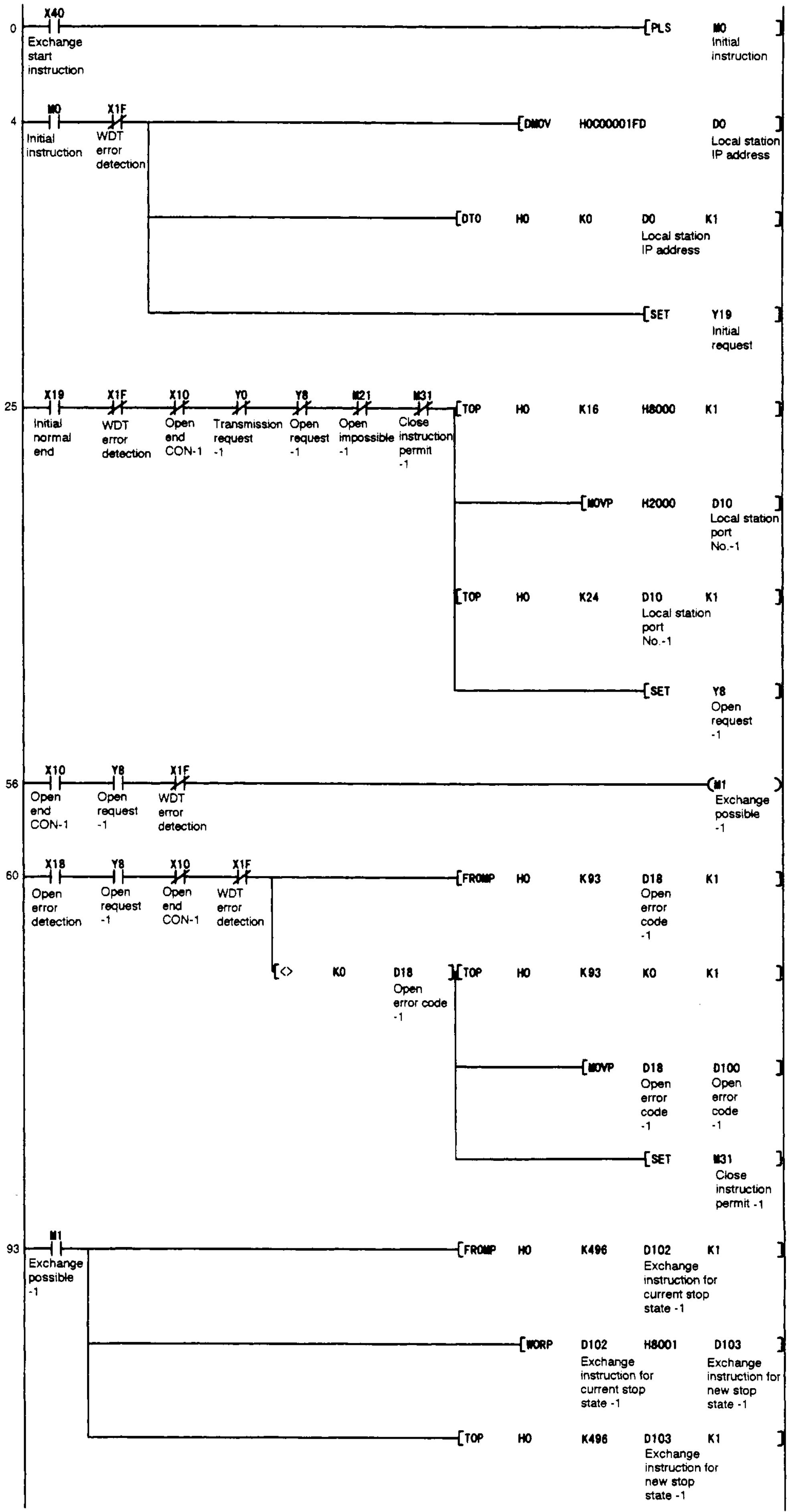
② Conducts open processing. (Exchange while the PLC CPU is stopped is made possible.) Two open processes are conducted to make possible exchange (transmission, reception) using a fixed buffer. In either case an Unpassive open is conducted and an Active open request from a remote node is awaited.

③ Exchange (transmission with procedure) using a fixed buffer, exchange using a random access buffer, and read/write data in the PLC CPU are conducted from the remote node. In addition, when not exchanging with the remote node, exchange (transmission with procedure) using a fixed buffer is conducted from the PLC CPU.

④ After data exchange is terminated, exchange will be prohibited while the PLC CPU is stopped and close processing will be conducted.

⑤ Termination processing is conducted after close processing is completed.

	Connection 1	Connection 2	Remark
E71 IP Address	C0. 00. 01. FD		Class= C, Net ID= 1, Host ID= FDH
E71 Port No.	2000H	2001H	Set the personal computer side to a free No.
Communication format	TCP/IP		Personal computer side is also TCP/IP
Open method	Unpassive	Unpassive	Personal computer side is Active open
Fixed buffer exchange	Transmission ready	Reception ready	———
Random access buffer exchange	Ready	Ready	Exchange is possible at either the connection 1 or connection 2.
Read/write data in the PLC CPU	Ready	Ready	

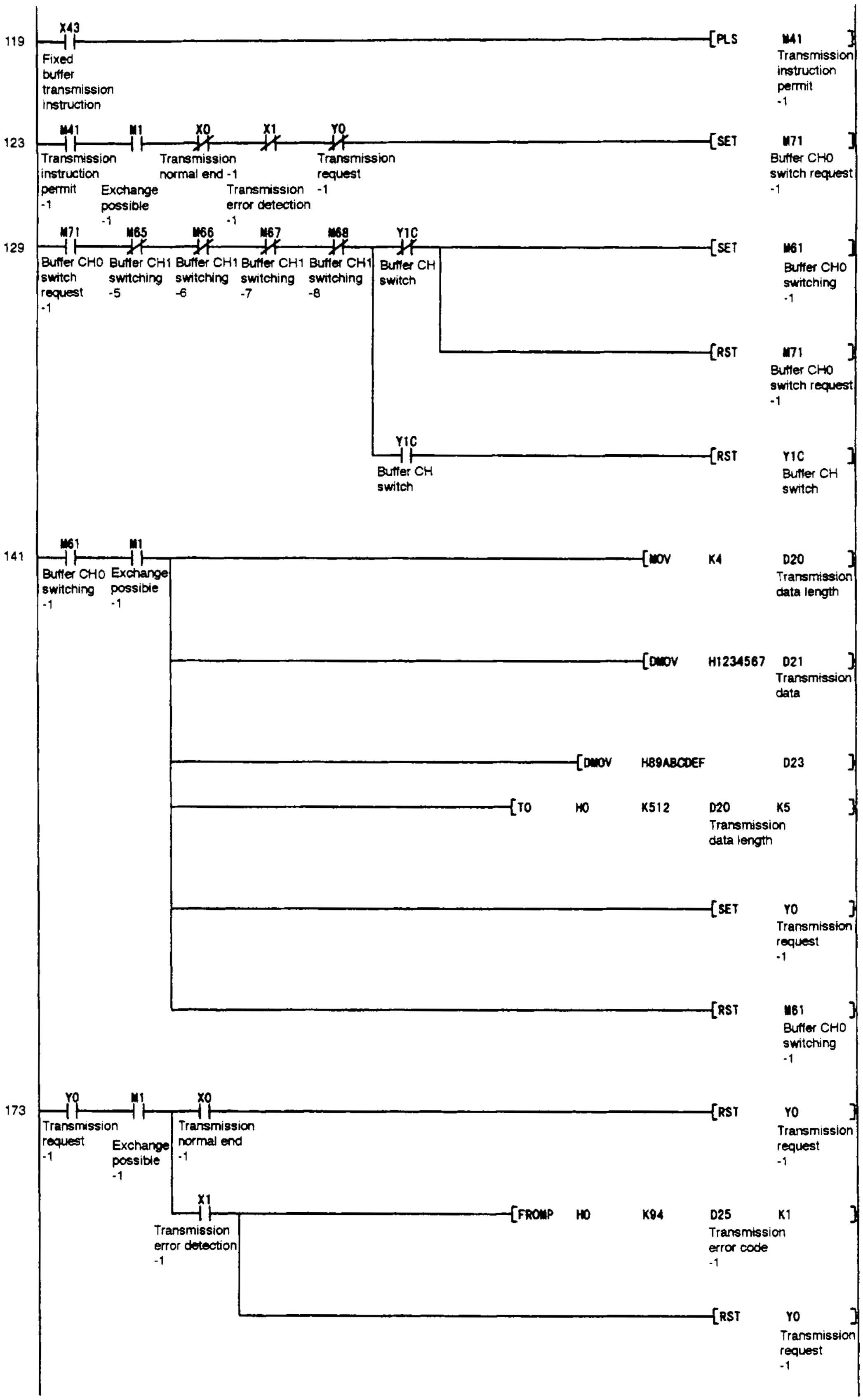


Initial processing.

Open processing (Connection 1)
- TCP/IP Unpassive open.
- Transmission using fixed buffer ready.
- Exchange using random access buffer ready.
- Read/write data in the PLC CPU ready.

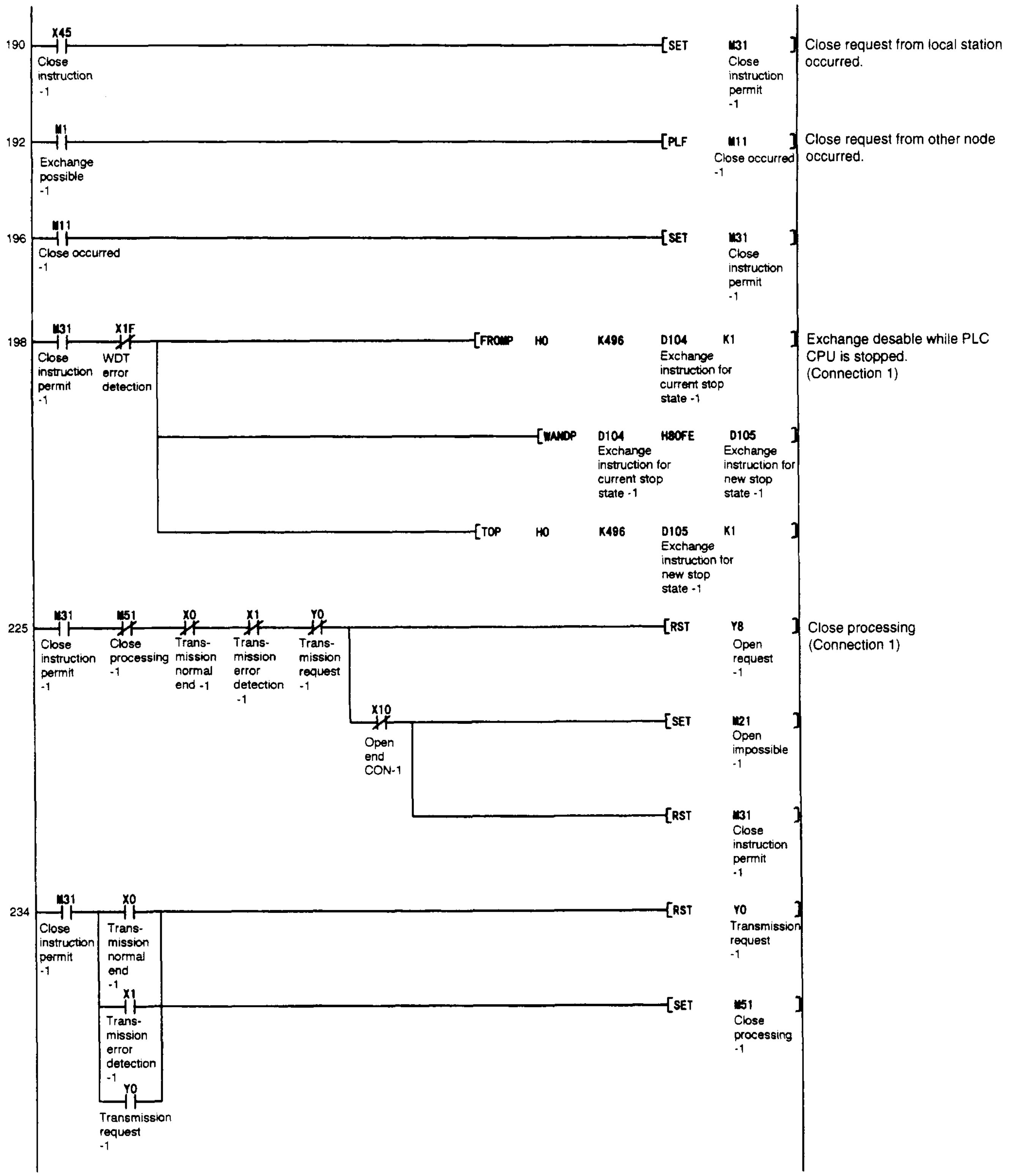
Open error processing. (Connection 1)

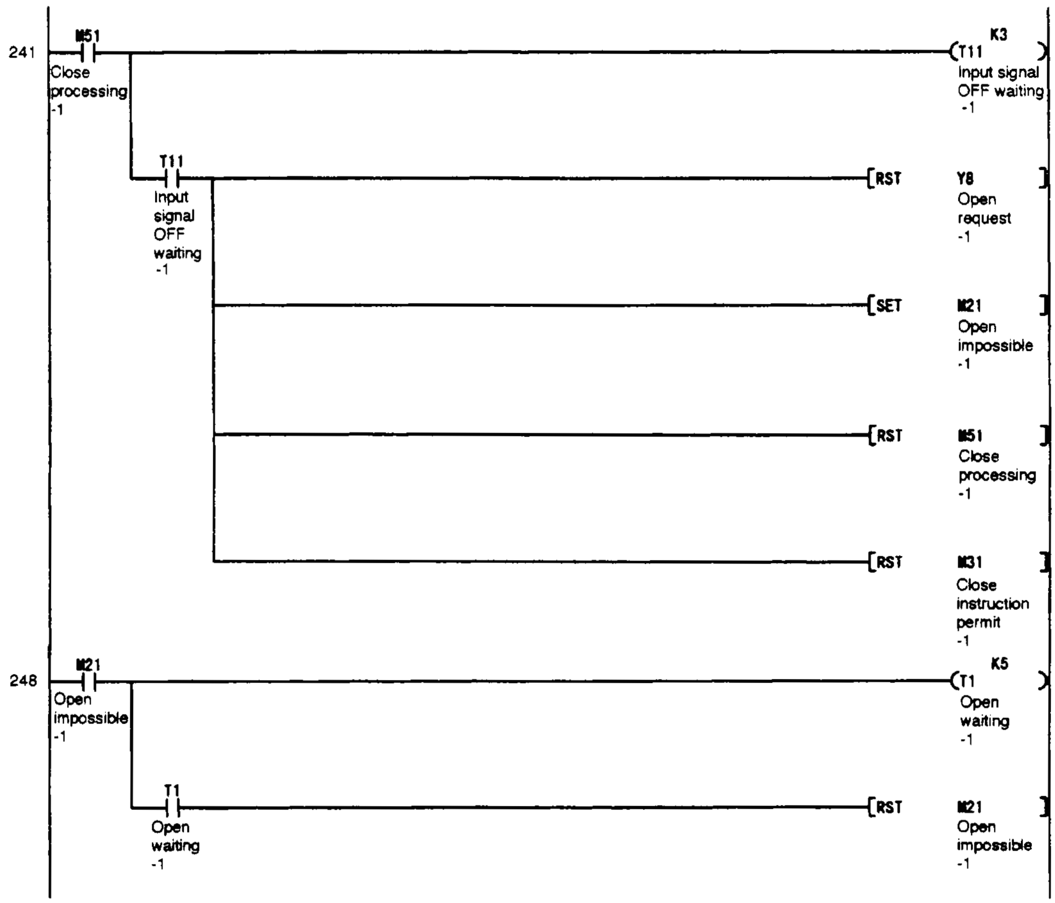
Exchange enabled while PLC CPU is stopped. (Connection 1)

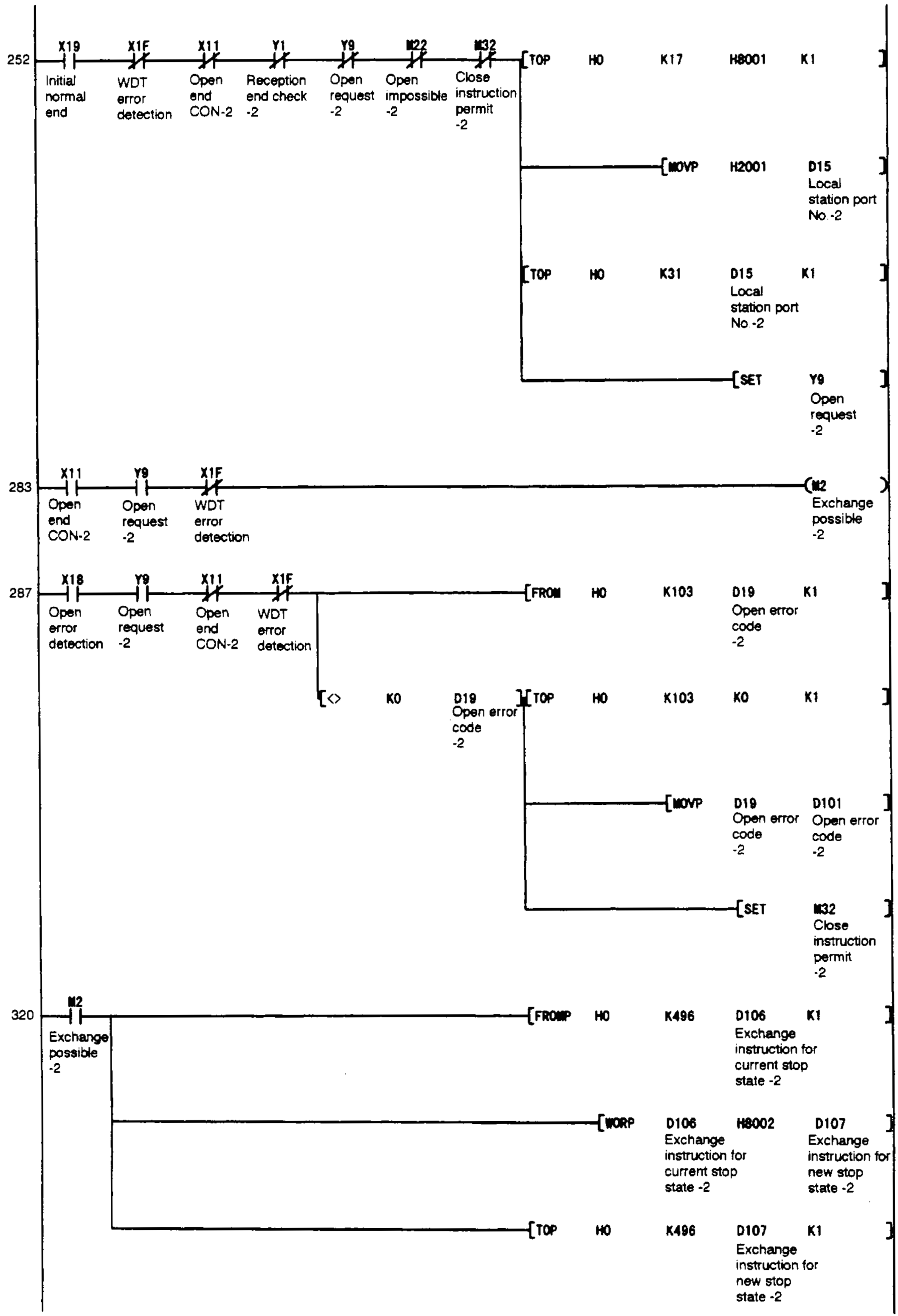


Fixed buffer transmission processing (Connection 1)

Channel switch of buffer memory completed.
Writing to the fixed buffer of connection 1 is possible.


190
X45
Close instruction -1
SET M31
Close instruction permit -1
Close request from local station occurred.
192
M1
Exchange possible -1
PLF M11
Close occurred -1
Close request from other node occurred.
196
M11
Close occurred -1
SET M31
Close instruction permit -1
198
M31
Close instruction permit -1
X1F
WDT error detection
FROMP H0 K496 D104 K1
Exchange instruction for current stop state -1
Exchange desable while PLC CPU is stopped. (Connection 1)
WANDP D104 H80FE D105
Exchange instruction for current stop state -1
Exchange instruction for new stop state -1
TOP H0 K496 D105 K1
Exchange instruction for new stop state -1
225
M31
Close instruction permit -1
M51
Close processing -1
X0
Trans-mission normal end -1
X1
Trans-mission error detection -1
Y0
Trans-mission request -1
RST Y8
Open request -1
Close processing (Connection 1)
X10
Open end CON-1
SET M21
Open impossible -1
RST M31
Close instruction permit -1
234
M31
Close instruction permit -1
X0
Trans-mission normal end -1
X1
Trans-mission error detection -1
Y0
Transmission request -1
RST Y0
Transmission request -1
SET M51
Close processing -1

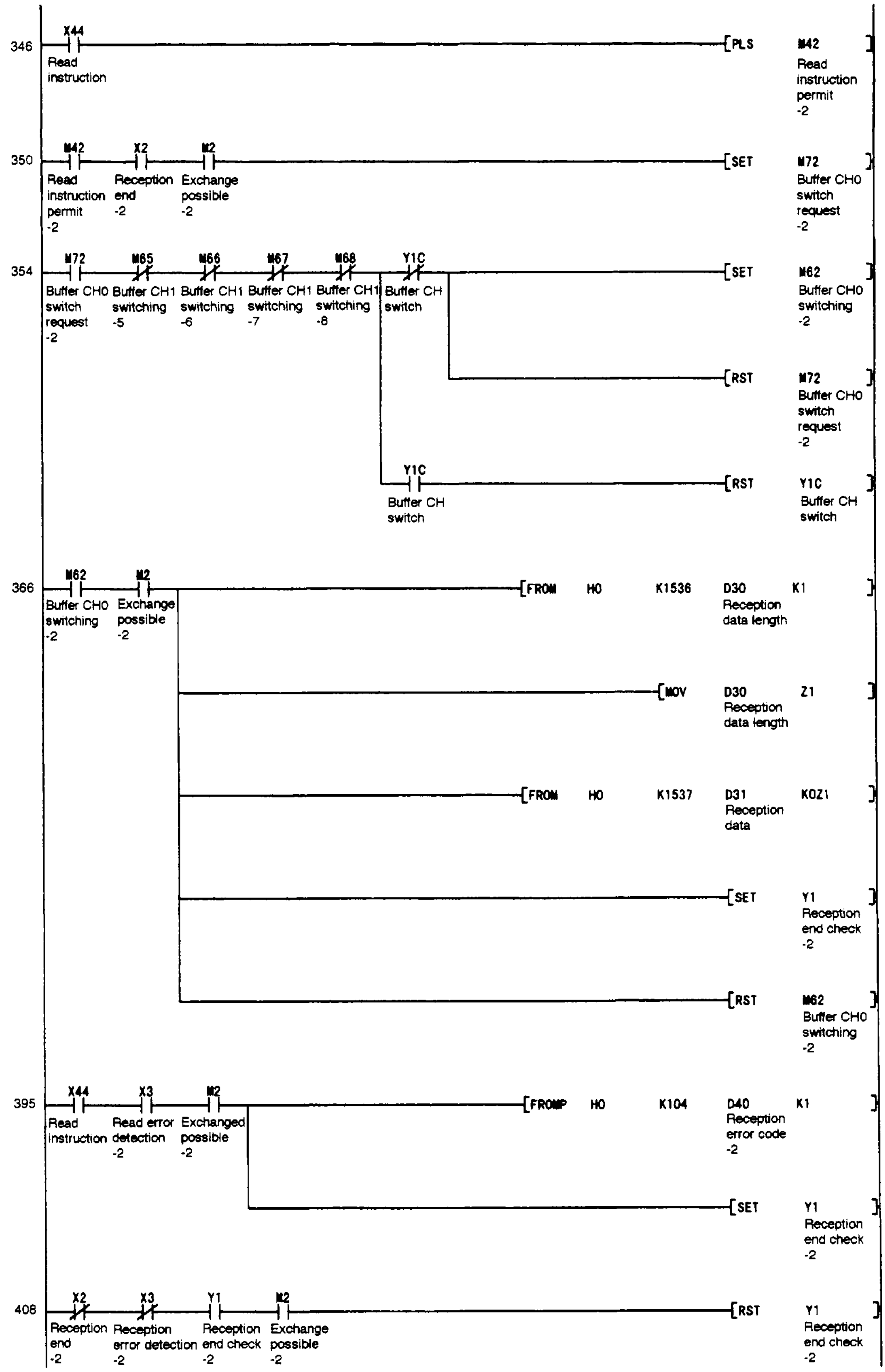

241
M51
Close processing -1
K3
T11
Input signal OFF waiting -1
T11
Input signal OFF waiting -1
RST
Y8
Open request -1
SET
M21
Open impossible -1
RST
M51
Close processing -1
RST
M31
Close instruction permit -1
248
M21
Open impossible -1
K5
T1
Open waiting -1
T1
Open waiting -1
RST
M21
Open impossible -1



Open processing.
(Connection 2)
- TCP/IP Unpassive open
- Reception using fixed buffer ready.
- Exchange using random access buffer ready.
- Read/write data in the PLC CPU ready.

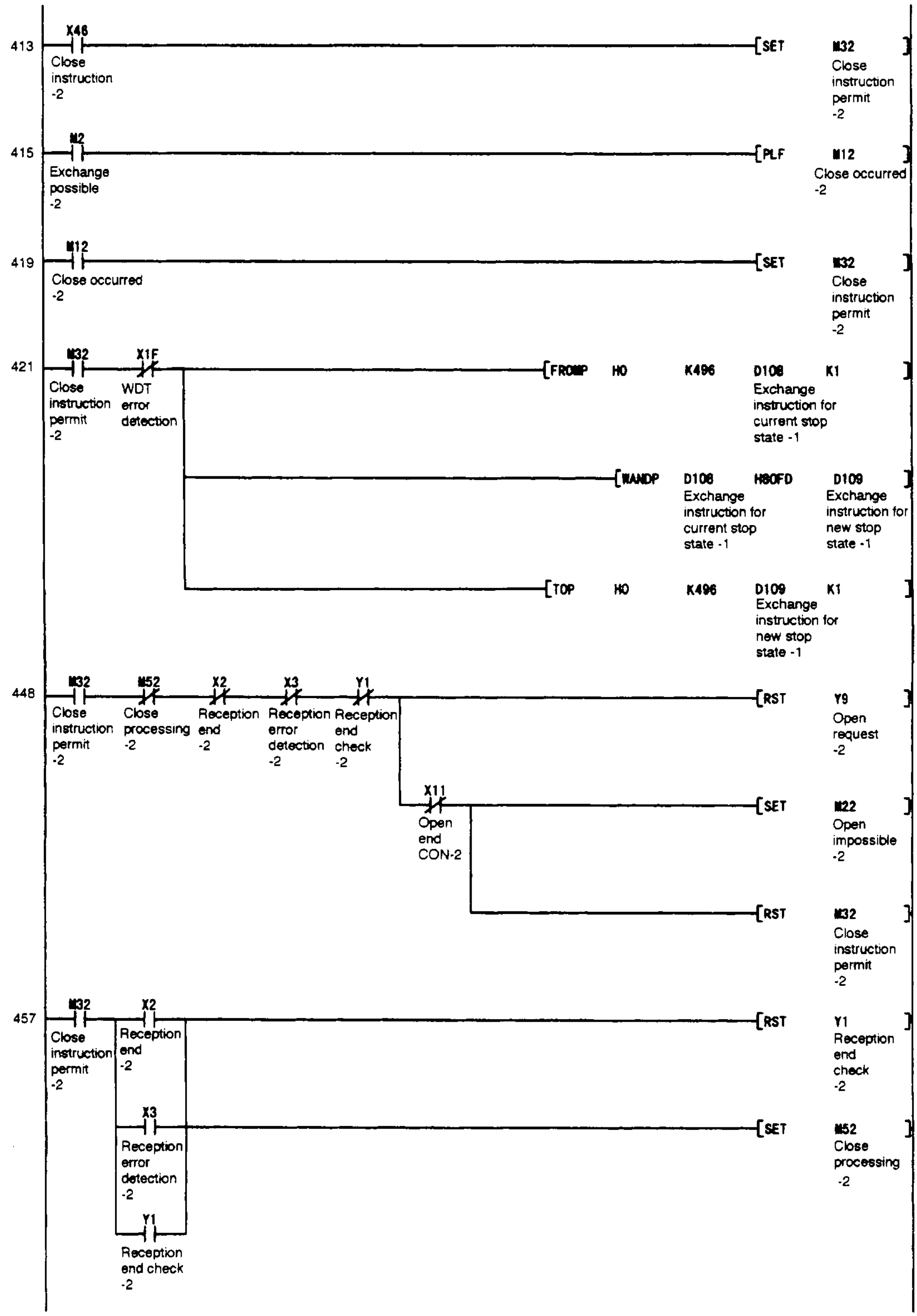
Open error processing.
(Connection 2)

Exchange enabled while PLC CPU is stopped.
(Connection 2)



Fixed buffer reception processing (Connection 2)

Channel switch of buffer memory completed.
Reading from the fixed buffer of connection 2 is possible.

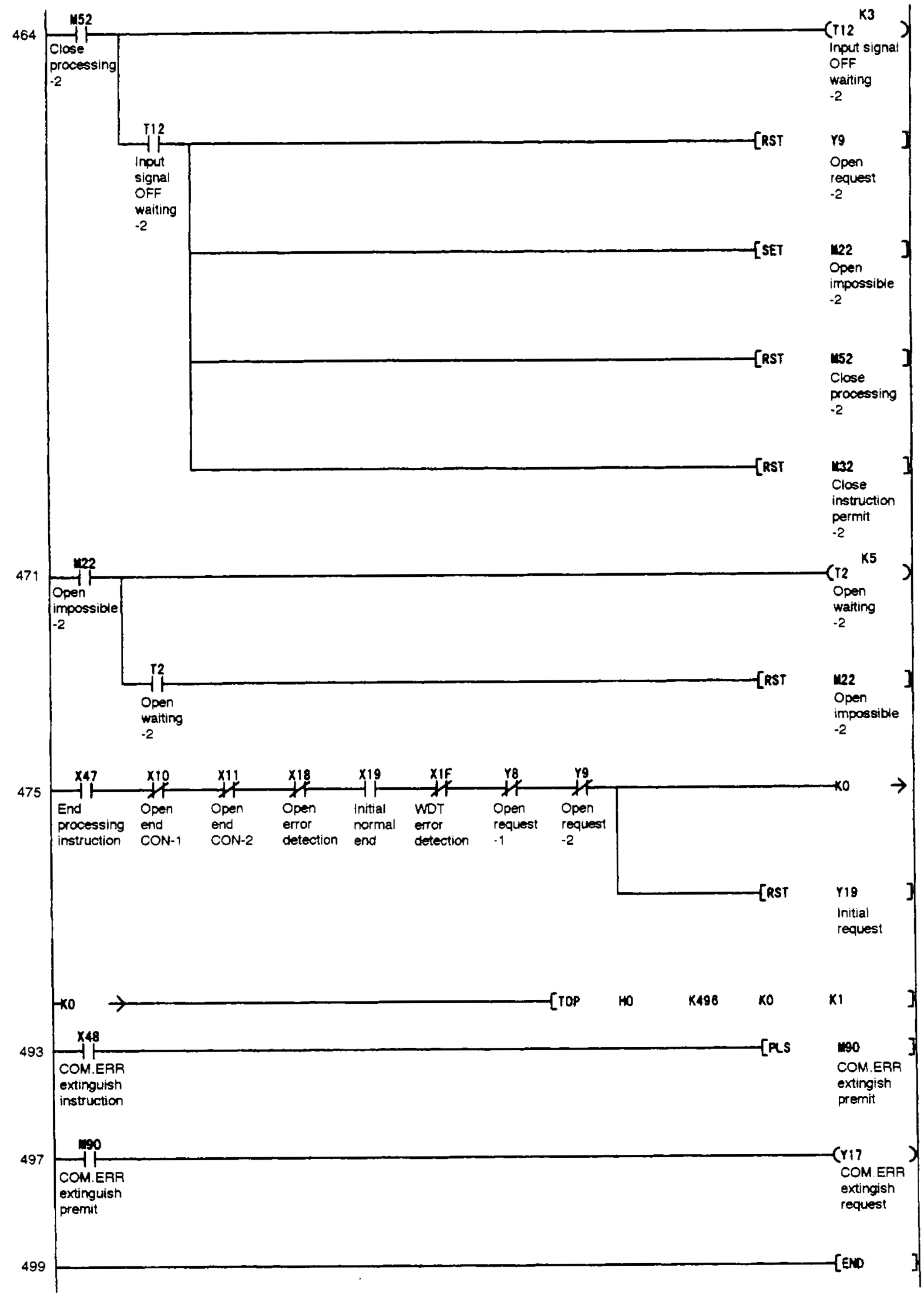


Close request from local station ocurred.

Close request from other node ocurred.

Exchange desable while PLC CPU is stopped.
(Connection 2)

Close processing
(Connection 2)



End processing.
Set all communication from PLC CPU in STOP as prohibited.

COM.ERR LED extinguish processing.

Appendix 7.3 Program for Reading/Writing Data in the PLC CPU

An example of the remote node side program that reads and writes data inside the PLC CPU is shown below. Also the sample program, execution environment, and data exchange are explained.

1 Sample program execution environment

① Except for the E71 IP Address, the setting values and switch settings on the PLC CPU side are the same as the execution environment shown in Appendix 7.1 Item 1 (a).

- E71 IP Address : 96.21.72.99H (150.33.114.153)

② Except for the following software development environment the node side has the same execution environment as that described in Appendix 7.1 Item 1 (b).

- Software development environment : Microsoft Corporation's visual basic (Ver.4.0) is used.
- The IP Address and port No. are allocated free Nos.

③ The communication format is TCP/IP.

2 Sample program overview

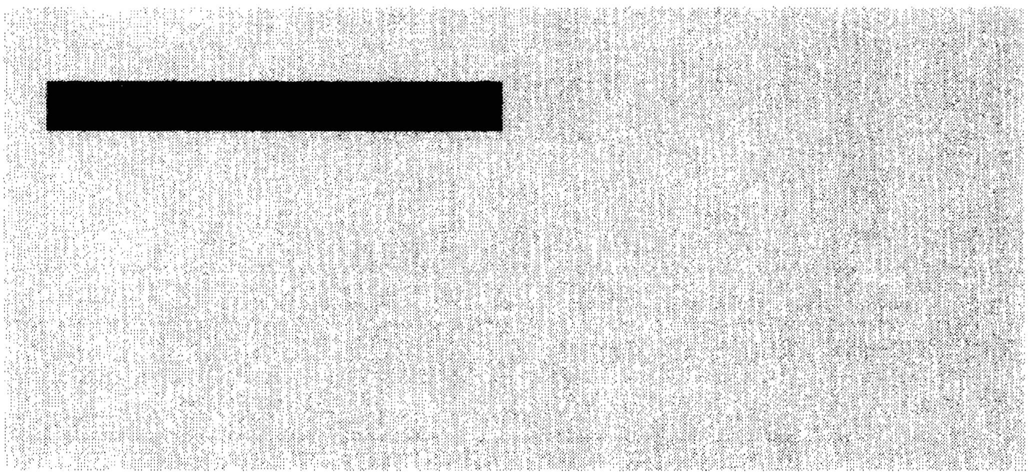
Data is written to D0 to D4 (5 points worth) of the AnUCPU installed station E71 by the E71 command (03: word unit batch write).

Executing the program in this item displays the message "Starting Up. Click Please" on the screen. Clicking the left button on the mouse will begin the data write to the AnUCPU. The program shown in Appendix 7.2 is used as the program for the PLC CPU side and the E71 IP address is changed to that above.

3 Screen setting file (Form1)

```
VERSION 4.00
Begin VB.Form Form1
   AutoRedraw      =   -1  'True
   Caption         =   "Form1"
   ClientHeight    =   3645
   ClientLeft      =   1395
   ClientTop       =   3660
   ClientWidth     =   7980
   Height          =   4050
   Left            =   1335
   LinkTopic       =   "Form1"
   ScaleHeight     =   3645
   ScaleWidth      =   7980
   Top             =   3315
   Width           =   8100
   Begin VB.Label Label1
      BackColor       =   &H00C0C000&
      Caption         =   "Starting Up.Click Please"
      Height          =   375
      Left            =   360
      TabIndex        =   0
      Top             =   600
      Width           =   3495
   End
End
Attribute VB_Name="Form1"
Attribute VB_Creatable=False
Attribute VB_Exposed=False
```

(Display contents when setting screen)



4 Main program (Form1)

```
Private Sub Form_Load()
    Dim r%
    r = ws_init()
    If (r <> 0)Then
        MsgBox "ws_init error" & r
    End If

End Sub
Private Sub Form_QueryUnload(cancel As Integer, UnloadMode As Integer)
    r= ws_term()
    If (r <> 0)Then
        MsgBox "ws_term error" & WSAGetLastError()
    End If
End Sub

Private Sub Label1_Click()
ssock = ws_start()
    If (ssock < 0) Then
        WSACleanup          'Release Winsock.DLL
        Form1.ScaleMode = 4'Specify character mode
        Form1.CurrentX = 4
        Form1.CurrentY = 12
        Form1.Print"Error Exit"
        Exit Sub
    End If
WSACleanup                  'Release Winsock.DLL
Form1.ScaleMode = 4         'Specify character mode
Form1.CurrentX = 4
Form1.CurrentY = 12
Form1.Print"Completed mormal"
End Sub
```

5 Subroutine program (Module 1)

```
Attribute VB_Name="Module1"
'Declare Sub Sleep Lib "kernel32" (ByVal dwMilliseconds As Long)
'----------------declares for Windows Socket API--------------------
'
Global Const WSADESCRIPTION_LEN = 256
Global Const WSASYS_STATUS_LEN = 128

Type WSADATA
   wVersion As Integer
   wHighVersion As Integer
   szDescription As String * 257    '(WSADESCRIPTION_LEN + 1)
   szSystemStatus As String* 129     '(WSASYS_STATUS_LEN + 1)
   iMaxSockets As Integer
   iMaxUdpDg As Integer
   lpVendorInfo As Long
End Type
'
Declare Function WSAStartup Lib "wsock32.dll"(ByVal wVerReq As Integer,lpWSAdata As WSADATA)
As Integer
Declare Function WSACleanup Lib "wsock32.dll"() As Integer
Declare Function WSAGetLastError Lib "wsock32.dll"() As Integer
'
'
Global Const SOCK_STREAM% = 1
Global Const SOCK_DGRAM% = 2
Global Const AF_INET% = 2
Global Const INADDR_ANY& = 0&

Type in_addr
    s_addr As Long
End Type
```

```
Type sockaddr_in
    sin_family As Integer
    sin_port As Integer
    sin_addr As in_addr
    sin_zero As String * 8
End Type
'
'system call
'
Declare Function bind Lib "wsock32.dll"(ByVal s As Integer, aname As sockaddr-in, ByVal
anamelen As Integer)As Integer
Declare Function closesocket Lib "wsock32.dll" (ByVal s As Integer) As Integer
Declare Function connect Lib "wsock32.dll"(ByVal s As Integer, aname As sockaddr_in, ByVal
anamelen As Integer)As Integer
Declare Function recv Lib "wsock32.dll"(ByVal s As Integer, ByVal buf As String, ByVal
buflen As Integer, ByVal flags As Integer) As Integer
Declare Function send Lib "wsock32.dll"(ByVal s As Integer, ByVal buf As String, ByVal
buflen As Integer, ByVal flags As Integer) As Integer
Declare Function socket Lib "wsock32.dll" (ByVal af As Integer, ByVal stype As Integer,
ByVal protocol As Integer)As Integer
'
' library
'
Declare Function htonl Lib "wsock32.dll" (ByVal hostlong As Long)As Long
Declare Function htons Lib "wsock32.dll" (ByVal hostshort As Integer)As Integer
Declare Function ntohl Lib "wsock32.dll" (ByVal netlong As Long)As Long
Declare Function ntohs Lib "wsock32.dll" (ByVal netshort As Integer)As Integer

Declare Function inet_addr Lib "wsock32.dll"(ByVal cp As String)As Long
'
'
Const ws_MAXMUD% = 1024
Global ws_data As WSADATA

Function ws_term%()
    MsgBox "Exit sample program"
End Function

Function ws_init%()
    Dim r%

    r = WSAStartup(&H101, ws_data)
    If (r = 0) Then
        If (ws_data.wVersion <> &H101)Then
            r = WSAVERNOTSUPPORTED
            If (WSACleanup() <> 0)Then
                r = WSAGetLastError()
            End If
        End If
    End If
    ws_init = r
End Function

Function ws_ start%()
'    Dim sock As Integer    '16 bitVB is integer
    Dim sock As Long        '32 bitVB is long
    Dim addr As sockaddr_in
    Dim s_in As sockaddr_in
    Dim s_buf$, r_buf$
'----Create soket----
    ws_start = -1
    sock = socket(AF_INET, SOCK_STREAM, 0)
    If (sock < 0) Then
        MsgBox "Socket error "& WSAGetLastError()
        Exit Function
    End If
```

```
'----Bind----
    s_in.sin_family = AF_INET
    s_in.sin_addr.s_addr = htonl(INADDR_ANY)    'Personal computer side
    s_in.sin_port = htons(0)                    'Personal computer side(port=1024 to 5000)
    If (bind(sock, s_in, Len(s_in)) <> 0)Then
        MsgBox "Bind error" & WSAGetLastError()
        Exit Function
    End If
'----Request connect(Active open)----
    addr.sin_family = AF_INET
    addr.sin_addr.s_addr = inet_addr("150.33.114.153")'PLC side
    addr.sin_port = htons(&H2000)                      'PLC side (port = 0X2000)
    If (connect(sock, addr, Len(addr))<> 0)Then
        MsgBox "Connect error "& WSAGetLastError()
            Exit Function
    End If
'----D0 to D4 Batch write request----
    s_buf$ = "03FF000A4420000000000500112233445566778899AA"
    r = send(sock, s_buf, Len(s_buf), 0)
    If (r <=0)Then
        Exit Function
    End If
'----Display transmission data----
    Form1.ScaleMode = 4            'Specify character mode.
    Form1.CurrentX = 4
    Form1.CurrentY = 5
    s_len$ = Len(s buf)
    Form1.Print"Transmission data =" +s_len$+ "byte"
    Form1.CurrentX = 8
    Form1.CurrentY = 6
    Form1.Print s_buf$
'
'----Reception response----
'   Call Sleep(100)
    Do
        r_buf = Space$(ws_MAXMUD)
        r = recv(sock, r_buf, LenB(r_buf), 0)
        If (r > 0) Then
            r_buf$ = Left$(r_buf, r)  'The character string portion of the received alpha-
                                       numeric portion is removed.
'
'----Display reception data----
                Form1.ScaleMode = 4       'Specify character mode.
                Form1.CurrentX = 4
                Form1.CurrentY = 8
                r_len$ = Len(r_buf)
                Form1.Print "response=" +r_len$ + "byte"
                Form1.CurrentX = 8
                Form1.CurrentY = 9
                Form1.Print r_buf$
                Exit Do
        End If
    Loop
'----Request close----
    ws_close = closesocket(sock)
'----Exit normal----
    ws_start = sock
End Function
```

Appendix 8 Difference between Ethernet and IEEE802.3

The Ethernet header for the data link layer supported by the E71 meets the specifications of the Ethernet frame.

The E71 does not communicate with another node whose Ethernet header for data link layer has the IEEE 802.3 (ISO/IEC 8802-3) specifications.

1 Ethernet

Destination address	Source address	Type	Data (46 to 1500 bytes)	FCS

2 IEEE802.3

Destination address	Source address	Length	Data (46 to 1500 bytes)	FCS

Appendix 9 E71 Support's ICMP Protocol

Shows the ICMP types and E71 processing supported by the E71.

ICMPType	ICMP Name/Description	E71 Processing
0	Echo Request IP packet echo request	When it receives an Echo Request, the E71 transmits an Echo Reply.
3	Destination Unreachable The IP packet could not reach the partner destination.	When data is received by a connection that has not been opened, the E71 transmits an error reply.
8	Echo Reply IP packet echo results	When the subject destination existence check is set in the buffer memory, the E71 transmits a command when the existence check is conducted.(*1)
Other	——	Ignored by the E71. (Not yet supported)

*1 The E71 can receive 2 ICMP ECHO requests (type 8) used for existence check, etc., at the same time and conducts the corresponding processing. When 3 or more ICMP ECHO requests are received at the same time the requests from the third and later are ignored. When an ICMP ECHO request is transmitted to the E71 from the remote node and a response is not returned to the remote node, retransmit the ICMP ECHO request to the E71.

Appendix 10 When Using the Ethernet Interface Module with a QnA Type PLC

The following lists limitations and notes when using the E71 with a QnACPU or remote system on the MELSECNET/10 network.

1 Availability of data communication function

The following table lists the availability of the E71 data exchange function with respect to the PLC to which the E71 is loaded and the stations accessed from other nodes.

E71 loaded station	Stations accessed from other nodes	Possibility of data exchange with other nodes		
		Read/write of data within the PLC CPU	Fixed buffer communication	Random access buffer communication
QnACPU station	Local station (E71 loaded station)	Communication is possible with the device memory within the AnACPU range.	Communication possible	Communication possible
	Other station (via MELSECNET/10)	Communication is possible with the QnACPU station device memory within the AnACPU range. For stations other than the QnACPU, all types of communication are possible within the specification range described in Chapter 9.	Communication not possible	Communication not possible
	Other station (via MELSECNET (II), MELSECNET/B)		Communication not possible	Communication not possible
MELSECNET/10 remote station	Local station (E71 loaded station)	Communication is possible with B, W, X, Y, special M9000's and special D9000's in the device memory as well as with special function modules.	Communication not possible	Communication not possible
	Other station (via MELSECNET/10)	Communication is possible with the QnACPU station device memory within the AnACPU range. For stations other than the QnACPU, all types of communication are possible within the specification range described in Chapters 9 and 10.	Communication not possible	Communication not possible

2 Maximum number of modules that can be loaded

A maximum of six E71 modules can be loaded to a single QnA type PLC. Follow the instructions provided in Item 2.2 to install the module.

3 Data read/write communication within the PLC CPU

Perform data communication after verifying the usable commands described in Item 9.2. Data read and write is possible with respect to the QnACPU device memory within the device range of AnACPU. (It is impossible to read and write the file register.)

4 Fixed buffer communication, random access buffer communication

Data exchange between the QnACPU and other nodes by means of the E71 is performed in the same manner as data exchange between the A-series PLC CPU and other nodes.

Point

Since the function specifications are different between the QnACPU and the A-series PLC CPU, the response speed from the PLC CPU will vary with respect to read/write requests.
When using a QnA type PLC with an E71 module installed, always verify that it operates normally.

Appendix 11 MELSEC Communication Support Software Tool

By using the Mitsubishi communication support software tool for supporting communication between the MELSEC-A Series and QnA Series Programmable Logic Controller (hereinafter PLC) and personal computer, the communication program on the personal computer side connected with Ethernet etc., can be simplified.

The following sections describe the outline functions and data link functions of the basic communication support tool (SWnD5F-CSKP-E, hereinafter CSKP). The programming methods on personal computer side for reading/writing data to the MELSEC PLC CPU using CSKP are also described.

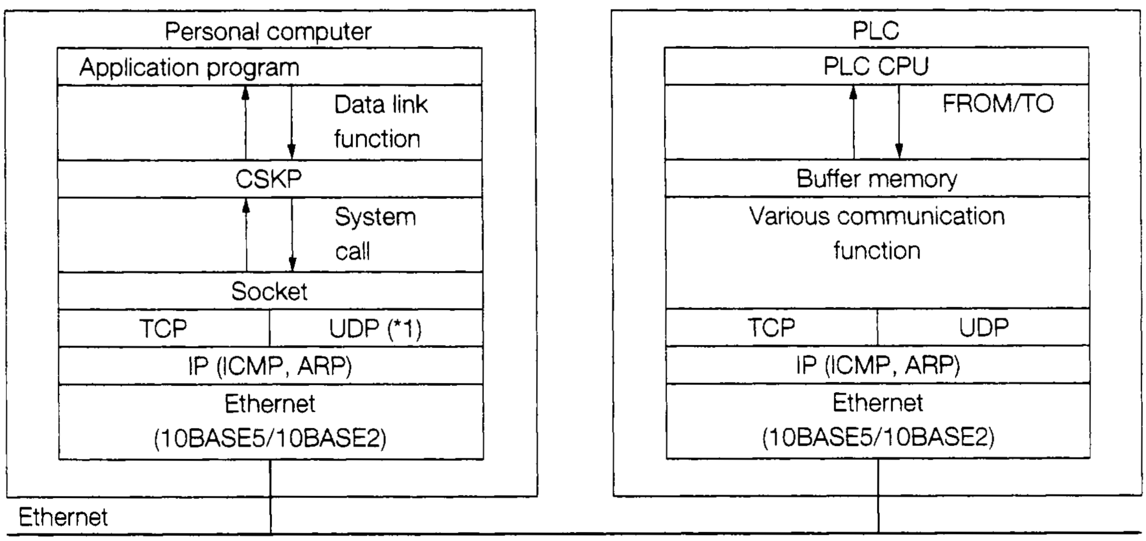
* For details on the CSKP, refer to the CSKP manual.

Appendix 11.1 Outline of basic communication support tool

This section shows the outline of CSKP.

1 **The CSKP is installed into the personal computer and used.**

(Example) Software structure diagram for Ethernet connection



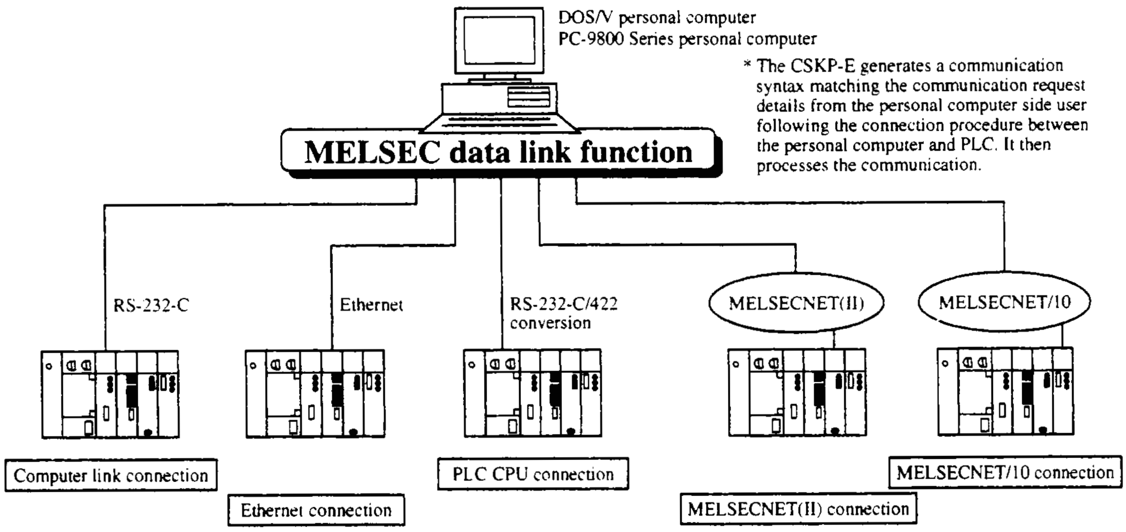
*1 The UDP/IP communication for using CSKP will be planned in the future.

2 **Ample support of communication protocols**

The personal computer side user can easily access the PLC's CPU by designating the connection type and communication request details using the following CSKP data link function arguments.

* The user does not need to understand the communication syntax unique to the special function module on the PLC side. The connection procedure can be changed just by changing part of the program (mdOpen argument's communication channel No.) in the program created by the user.

  Once created, the user program can be used effectively.



3 Data link functions dedicated for communicating data with PLC are supported

- Functions required for communicating data with the PLC, such as communication line open/close and device read/write, are prepared as data link functions.
- Various communication programs can be created easily, so development of the communication system using Ethernet/serial communication, etc. with the PLC side is easily.

Function name	Data link function's function
mdOpen	Initializes and opens the designated communication line channel.
mdClose	Closes the designated communication line channel.
mdSend	Writes the data in a batch to the designated device.
mdReceive	Reads the data in a batch from the designated device.
mdRandR	Reads data from a randomly designated device.
mdRandW	Writes the data to a randomly designated device.
mdDevSet	Sets (ON) the designated bit device.
mdDevRst	Resets (OFF) the designated bit device.
mdControl	Carries out remote RUN/STOP/PAUSE in respect to the designated PLC CPU.
mdTyperead	Reads the type of designated PLC CPU.

4 Various utilities are enclosed

The communication between the PLC and personal computer can be tested and monitored with the communication diagnosis utility and device monitor utility enclosed with the product.

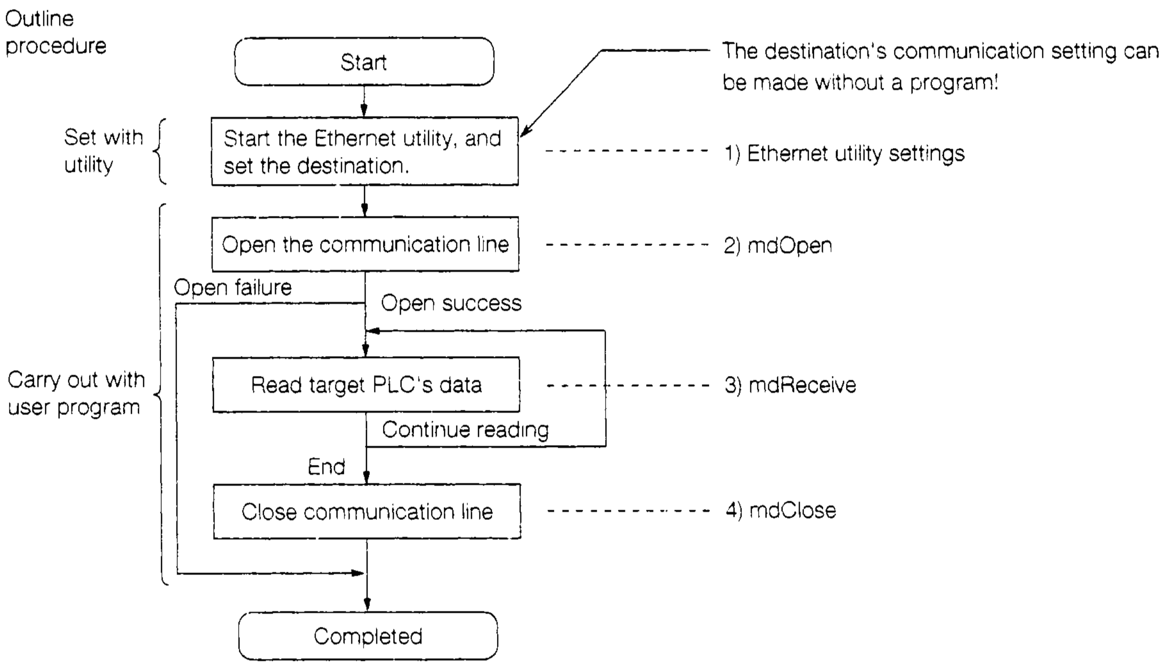
Appendix 11.2 Examples of using basic communication support tool

Programming procedures and program examples for connecting to the PLC's Ethernet interface module and computer link module and accessing the PLC's CPU from a personal computer are described in this section.

For the program example, the personal computer reads the data register (D) in the A/QnA Series PLC CPU, and displays the results on the screen.

1 Programming procedures (Outline flow of procedures)

The flow of the data read process for creating a personal computer side program to access the PLC CPU using CSKP is shown below.



(1) Ethernet utility settings

Set the module type (E71, QE71), host name (IP address) and port No., etc., of the PLC to be communicated with over an Ethernet connection. This information is saved as a logic station No.

(2) mdOpen

This function initializes and opens the Ethernet communication line channel (No. = 61).

The path of the opened communication line is returned as the execution results.

The path is required to execute the function (mdReceive) for reading the PLC data, and the function (mdClose) for closing the line.

(3) mdReceive

This function reads the PLC CPU's device data.

The target PLC's logic station No., device to be read and No. of points are assigned to the argument.

(4) mdClose

This function closes the opened communication line channel.

The communication line must be closed when reading and writing are completed.

2 Programming example (Using Visual C++)

The D0 and D1 values of the target PLC CPU are read over the logic station No. 0 communication line, and are displayed on the screen.

```
/********************************************
* Include                                   *
********************************************/
#include    <stdio.h>
#include     <windows.h>          /* Windows include file    */
#include     "mdfunc.h"      /* Data link function include file */
/********************************************
* Constant declaration                      *
********************************************/
#define CHSN_ETHERNET        61    /* Ethernet communication channel (common
for E71, QE71)*/
#define MODE_DUMMY      -1    /* Dummy (fixed to -1)              */
#define STNO_ROGIC           0     /* Logic station No.           */
#define DEVTYPE_D            13    /* Device type "D"             */
#define DEVNO_0              0     /* Read top device No.              */
#define SIZE_R_BYTES         4     /* No. of read data bytes      */
#define DATA_INITIAL         0     /* Read area initial value     */
/****************************************************
* The logic station No. "0" PLC data is read.       *
* The logic station No. is preset with the Ethernet utility.     *
****************************************************/
void main( )
{
   /* Declare mdOpen argument      */
   long path;                      /* Path save function              */
   short chan;                     /* Communication line channel No. */
   short mode;                     /* Dummy (fixed to -1)             */
   short oret;                     /* mdOpen function return value    */
   /* Declare mdReceive argument   */
   short stno;                     /* Station No.                     */
   short devtyp;                   /* Device type              */
   short  devno;                   /* Read top device No.      */
   short  size;                    /* No. of read data bytes     */
   short  data[2];                 /* Read data buffer           */
   short  rret;                    /* mdReceive function return value  */
   /* Declare mdClose argument     */
   short  cret;                    /* mdClose function return value    */
   /* Declare mdOpen argument setting  */
   chan  = CHSN_ETHERNET;          /* Ethernet communication channel   */
   mode  = MODE_DUMMY;             /* Dummy (fixed to -1)              */
   /* Execute mdOpen and open line  */
   oret = mdOpen( chan, mode, &path );
   if( oret != 0 ){
```

```
        mode  = MODE_DUMMY;                /* Dummy (fixed to -1)                    */
        /* Execute mdOpen and open line  */
        oret = mdOpen( chan, mode, &path );
        if( oret != 0 ){
                /* If function fails, return error        */
                printf( "mdOpen error[%04x]\n", oret );
        }else{
                /* Continue process only when mdOpen succeeds. */
                /* Set mdReceive argument          */
                stno    = STNO_ROGIC;          /* Logic station No.                   */
                devtyp = DEVTYPE_D;            /* Device type "D"                      */
                devno  = DEVNO_0;              /* Read device No. 0, 1                              */
                size    = SIZE_R_BYTES;        /* 2-point, 4-byte (word device is 1-point 2-byte)*/
                data[0] = DATA_INITIAL;        /* Initialize read data area     */
                data[1] = DATA_INITIAL;        /* Initialize read data area     */
                /* Execute mdReceive and read          */
                /* Use path acquired with mdOpen         */
                rret = mdReceive( path, stno, devtyp, devno, &size, &data );
                if( rret != 0 ){
                        /* If function fails, return error       */
                        printf( "mdReceive error[%04x]\n", rret );
                }else{
                        /* Display read data               */
                        printf( "data0 [%4x]\n", data[0] );    /* Data display*/
                        printf( "data1 [%4x]\n", data[1] );    /* Data display*/
                }
                /* Execute mdClose and close line        */
                /* Use path acquired with mdOpen         */
                cret = mdClose( path );
                if( cret != 0 ){
                        /* If function fails, return error       */
                        printf( "mdClose error[%04x]\n", cret );
                }
        }
}
```

Appendix 12 Index

(P)

(Q)

(R)

(S)

(T)

(U)

(1)

WARRANTY

Please confirm the following product warranty details before starting use.

1. Gratis Warranty Term and Gratis Warranty Range

If any faults or defects (hereinafter "Failure") found to be the responsibility of Mitsubishi occurs during use of the product within the gratis warranty term, the product shall be repaired at no cost via the dealer or Mitsubishi Service Company. Note that if repairs are required at a site overseas, on a detached island or remote place, expenses to dispatch an engineer shall be charged for.

[Gratis Warranty Term]

The gratis warranty term of the product shall be for one year after the date of purchase or delivery to a designated place.
Note that after manufacture and shipment from Mitsubishi, the maximum distribution period shall be six (6) months, and the longest gratis warranty term after manufacturing shall be eighteen (18) months. The gratis warranty term of repair parts shall not exceed the gratis warranty term before repairs.

[Gratis Warranty Range]

(1) The range shall be limited to normal use within the usage state, usage methods and usage environment, etc., which follow the conditions and precautions, etc., given in the instruction manual, user's manual and caution labels on the product.

(2) Even within the gratis warranty term, repairs shall be charged for in the following cases.

1. Failure occurring from inappropriate storage or handling, carelessness or negligence by the user. Failure caused by the user's hardware or software design.
2. Failure caused by unapproved modifications, etc., to the product by the user.
3. When the Mitsubishi product is assembled into a user's device, Failure that could have been avoided if functions or structures, judged as necessary in the legal safety measures the user's device is subject to or as necessary by industry standards, had been provided.
4. Failure that could have been avoided if consumable parts (battery, backlight, fuse, etc.) designated in the instruction manual had been correctly serviced or replaced.
5. Failure caused by external irresistible forces such as fires or abnormal voltages, and Failure caused by force majeure such as earthquakes, lightning, wind and water damage.
6. Failure caused by reasons unpredictable by scientific technology standards at time of shipment from Mitsubishi.
7. Any other failure found not to be the responsibility of Mitsubishi or the user.

2. Onerous repair term after discontinuation of production

(1) Mitsubishi shall accept onerous product repairs for seven (7) years after production of the product is discontinued. Discontinuation of production shall be notified with Mitsubishi Technical Bulletins, etc.

(2) Product supply (including repair parts) is not possible after production is discontinued.

3. Overseas service

Overseas, repairs shall be accepted by Mitsubishi's local overseas FA Center. Note that the repair conditions at each FA Center may differ.

4. Exclusion of chance loss and secondary loss from warranty liability

Regardless of the gratis warranty term, Mitsubishi shall not be liable for compensation to damages caused by any cause found not to be the responsibility of Mitsubishi, chance losses, lost profits incurred to the user by Failures of Mitsubishi products, damages and secondary damages caused from special reasons regardless of Mitsubishi's expectations, compensation for accidents, and compensation for damages to products other than Mitsubishi products and other duties.

5. Changes in product specifications

The specifications given in the catalogs, manuals or technical documents are subject to change without prior notice.

6. Product application

(1) In using the Mitsubishi MELSEC programmable logic controller, the usage conditions shall be that the application will not lead to a major accident even if any problem or fault should occur in the programmable logic controller device, and that backup and fail-safe functions are systematically provided outside of the device for any problem or fault.

(2) The Mitsubishi general-purpose programmable logic controller has been designed and manufactured for applications in general industries, etc. Thus, applications in which the public could be affected such as in nuclear power plants and other power plants operated by respective power companies, and applications in which a special quality assurance system is required, such as for Railway companies or National Defense purposes shall be excluded from the programmable logic controller applications.
Note that even with these applications, if the user approves that the application is to be limited and a special quality is not required, application shall be possible.
When considering use in aircraft, medical applications, railways, incineration and fuel devices, manned transport devices, equipment for recreation and amusement, and safety devices, in which human life or assets could be greatly affected and for which a particularly high reliability is required in terms of safety and control system, please consult with Mitsubishi and discuss the required specifications.

Ethernet Interface Module
type AJ71E71-S3, A1SJ71E71-B2-S3, A1SJ71E71-B5-S3

User's Manual

MODEL	Ethernet-U-S-E
MODEL CODE	13J856
SH(NA)3598-E(0006)MEE	

MITSUBISHI ELECTRIC CORPORATION

HEAD OFFICE : MITSUBISHI DENKI BLDG MARUNOUCHI TOKYO 100-8310 TELEX : J24532 CABLE MELCO TOKYO
NAGOYA WORKS : 1-14 , YADA-MINAMI 5 , HIGASHI-KU, NAGOYA , JAPAN